

XML API, version 15

Release 2024:52

Document version: 2024-08-07

The API contains two different search methods that can be implemented, static or non-static. We recommend implementing the non-static as this will result in better performance (lower response times, less data transferred), and it is also the expected solution for any customer with 250 requests/hour or more (peak load over 1 hour, not average over 24 hours). Static endpoint can only be used if previously agreed with Sunhotels.

Table of contents

Table of contents	2
Change log.....	3
Introduction.....	7
Global errors.....	8
Searching	10
Search V.3.....	23
PreBook.....	32
PreBook V.2.....	35
PreBook V.3.....	39
Booking.....	41
Book V.3	48
Retrieving room type list	51
Checking price difference for an amendment.....	52
Booking an amendment	54
Cancelling a booking.....	56
Retrieving a list of transfer types.....	58
Retrieving a list of themes.....	59
Retrieving a list of destinations	60
Retrieving a list of resorts.....	62
Transfer searching.....	64
Transfer searching V2.....	66
Retrieving a list of static hotels and rooms.....	69
Retrieving booking information.....	74
Retrieving booking information V2	75
Retrieving booking information V3	76
Retrieving transfer booking information.....	77
Booking a transfer.....	80
Booking a transfer V2.....	84
Cancelling a transfer booking.....	88
Downloading static data.....	90
Country, Destination and Resort notes	95
Retrieving all supported languages.....	96
Retrieving all meals.....	97
Retrieving all features	98
Retrieving all hotel note descriptions	99
Retrieving all room note descriptions.....	100
Retrieving booking special request	101
Updating booking special request	102

Change log

This section describes all changes made to the API since v14, function by function.

New functions

AmendmentPriceRequest/ AmendmentRequest

Enables changing the stay length (add/remove days) by automatically re-booking or making an additional booking for the added dates.

SearchV2 and BookV2

Replace old Search and Book methods. Include same existing parameters, also *customerCountry* and *b2c*, so far available through MarketSearch endpoint (now deprecated). Add the functionality of informing children ages, between 2 and 17, which is used to calculate the proper price and disregard adult only properties. Available through StaticXMLAPI and NonStaticXMLAPI endpoints.

PreBook

To be used in the context of a 3 step booking process, after the first search and before the booking is confirmed. Only valid together with SearchV2 and BookV2 versions.

PreBook V2

Same as the PreBook but with the functionality of pricebreakdown. Only valid together with SearchV2 and BookV2 versions.

GetBookingInformation V2

Same as the GetBookingInformation but with the functionality of showGuests. Allows to return guests names using the showGuests parameter.

SearchTransfers V2

Same as the SearchTransfer but with the functionality of giataCode. Allows to search a transfer using the giata code of a given property.

AddTransfer V2

Same as the AddTransfer but including more options to create a transfer booking. Additionally, a departureIataCode is now required to create arrival transfer bookings.

SearchV3, PreBookV3, BookV3 and GetBookingInformationV3

Same as previous versions but with the functionality of taxes and fees. Available through StaticXMLAPI and NonStaticXMLAPI endpoints.

Global changes

Search (**Please use SearchV2 instead)

Input

The *"numberOfInfants"* element has been renamed to *"infant"*.

A *"paymentMethodId"* parameter has been added, which specifies the payment method all rooms must have.

A *"currency"* parameter has been renamed to *"currencies"* and now specifies all the currencies that prices and discounts should be expressed in.

A *"childrenAges"* parameter has been added, which allows informing children ages, separated by comma if several.

A *"customerCountry"* parameter has been added, which allows informing the customer nationality.

A *"b2c"* parameter has been added, to specify whether the booking comes from a b2b/package or b2c point of sale

Output

The *"isPrioritized"* element has been removed and replaced with an attribute to the hotel.

The *"codes"* element has been added, which contains *"code"* elements. Each *"code"* element has two attributes, *"type"* and *"value"*, which specifies the type of code and the code value.

The “*distance*” element has been added, which specifies the distance between the hotel and the reference point when searching with coordinates.

The “*paymentMethods*” element has been added to the “*room*” element. This element contains all available payment methods for the room. Each payment method may also contain any number of properties.

The “*price_in_EUR*” and “*price*” elements have been replaced with the new “*prices*” element, which contains a list of prices expressed in all of the provided currencies.

The “*discountInEuros*” and “*discountInLocalCurrency*” elements under the “*discount*” element have been replaced with the new “*amounts*” element, which contains a list of discount amounts expressed in all of the provided currencies.

GetStaticHotelsAndRooms

Output

The “*codes*” element has been added, which contains “*code*” elements. Each “*code*” element has two attributes, “*type*” and “*value*”, which specifies the type of code and the code value.

The “*paymentMethods*” element was added to the “*room*” element. This element contains all available payment methods for the room, as well as the properties for each payment method.

GetRoomTypes

Book (**Please use BookV2 instead)

Input

The “*infants*” element has been renamed to “*infant*”.

A “*paymentMethodId*” parameter was added, which specifies the payment method a booking should be booked with.

Credit card information parameters were also added and are required when booking a Pay at Hotel room.

An “*invoiceRef*” parameter was added. This element contains an invoice reference.

A “*customerCountry*” parameter has been added, which allows informing the customer nationality.

A “*b2c*” parameter has been added, to specify whether the booking comes from a b2b/package or b2c point of sale

A “*commissionAmountInHotelCurrency*” parameter has been added, to be used together with “pay at hotel” payment method.

Only latin alphabet characters are allowed to use in it.

Output

The “*boardId*”, “*board*” and “*englishBoard*” elements has been renamed to “*mealId*”, “*meal*” and “*englishMeal*”.

The “*boardId*” element has been added, which specifies the id of the booked meal.

The “*price*” and “*price.EUR*” elements have been replaced with the new “*prices*” element, which contains the prices for the booking expressed in both the booked currency as well as euro.

The element “*paymentmethod*” has been added. This element contains a description of the booked payment method, along with any properties it has.

The “*voucherText*” element has been removed from the “*booking*” element and replaced with the new “*bookedBy*” element.

The “*voucherURL*” element has been renamed to “*voucher*”.

The element “*boardLabel*” has been added. This element contains additional board information.

The element “*englishBoardLabel*” has been added. This element contains additional board information (in English).

GetFlagTypes

GetDestinations

GetResorts

SearchTransfers

Input

The “*currency*” parameter has been renamed to “*currencies*” and now specifies all the currencies that rate prices should be expressed in.

Output

The “*priceEUR*” and “*price*” elements have been replaced with the new “*prices*” element, which contains a list of rate prices expressed in all of the provided currencies.

SearchRooms

GetBookingInformation

Output

The `"boardId"`, `"board"`, `"boardLabel"`, `"englishBoard"` and `"englishBoardLabel"` elements has been renamed to `"mealId"`, `"meal"`, `"mealLabel"`, `"englishMeal"` and `"englishMealLabel"`.

The `"boardId"` element has been added, which specifies the id of the booked meal.

The `"price"` and `"price.EUR"` elements have been replaced with the new `"prices"` element, which contains the prices for the booking expressed in both the booked currency as well as euro.

The element `"paymentMethod"` has been added. This element contains a description of the booked payment method, along with any properties it has.

The `"voucherText"` element has been removed from the `"booking"` element and replaced with the new `"bookedBy"` element.

The `"voucherURL"` element has been renamed to `"voucher"`.

The element `"boardLabel"` has been added. This element contains additional board information.

The element `"englishBoardLabel"` has been added. This element contains additional board information (in English).

The element `"invoiceRef"` has been added. This element contains an invoice reference.

GetTransferBookingInformation

Output

The element `"yourRef"` has been added. This element contains an internal reference.

The element `"invoiceRef"` has been added. This element contains an invoice reference.

AddTransfer

An `"yourRef"` parameter has been added. This element contains an internal reference.

An `"invoiceRef"` parameter has been added. This element contains an invoice reference.

"When using SOAP the arrival date and arrival time is sent using one parameter `"arrivalTime"` containing both date and time."

"When using SOAP the return date and return time is sent using one parameter `"returnDepartureTime"` containing both date and time."

CancelTransferBooking

GetTransferTypes

Important note

In order to simplify the reading of the documentation, the references to old Search and Book methods, available through XMLAPI and MarketSearch endpoints have been removed from this document, as they are considered deprecated and replaced by SearchV2 and BookV2.

Introduction

Starting at version 13 of the Sunhotels XML API, the solution will be WSDL based, which is a quite big difference to previous versions. It will also introduce a lot of new features, especially when you look at the search method. Many of our customers will appreciate that all available meals and meal prices is included directly in the search response. You can also find information about cancellation rules and airport transfers in the search result. We believe that this will lower the number of round trips needed to present a good search result to your customers.

Static / Non static search

The hotel search will be available in two versions. One will send static data that isn't likely to change, and the other will leave that information out. Examples of static data is hotel name, hotel description and transfer description. We recommend all customers to use the non-static search if possible. This will lower the amount of data transferred to you and will increase the response time. Look at the documentation for "DownloadExport" for more information about how to fetch our static data so you can cache it at your side. The cache is also useful if you have to map Sunhotels hotels to the hotels in your system.

SOAP / POST / GET

Version 13 of Sunhotels XML API will be available using POST /GET or SOAP. SOAP will be the recommended protocol to use from now on. The functionality will however not differ depending of protocol. We have decided to expose the POST/GET and the SOAP interfaces on two different pages. Most of the documentation below will be based on the response from a GET request despite the fact that we recommends SOAP since it's easier to understand the XML. It should however not be hard to understand how the SOAP response will work since it basically uses the same structure.

Correct URL for the web service

There exist four public web services that basically does the same work but is adjusted to if you want to make POST / GET / SOAP requests or if you want Static / Non static response.

	POST / GET	SOAP
Show static data	/15/PostGet/StaticXMLAPI.asmx	/15/SOAP/StaticXMLAPI.asmx
Do not show static data	/15/PostGet/NonStaticXMLAPI.asmx	/15/SOAP/NonStaticXMLAPI.asmx

If you for example want to search hotels and POST data and want to receive non static data you should make the request to: <http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx>

Compression

We will compress the data we send as response from version 13 and forward. We will look for the request header named "Accept-Encoding" and see if it contains gzip or deflate. Make sure your HTTP client doesn't send this to us if you don't wish to receive compressed content from Sunhotels.

Compression will give a bit CPU overhead but since XML isn't a very compact format it will give a boost when transferring over internet.

Global errors

Listed below are common errors that may be returned under the “Error” element in the XML output for all functions.

Error Type	Description	Resolution
CheckOutDatePrecedesCheckInDateException	The provided check-out date precedes the provided check-out date.	Change the check-out date to a date later than the check-in date.
DateCannotBeInPastException	A date has been provided for a parameter that does not accept dates in the past.	Change the date to a future date.
InconsistentParameterValuesException	The value of a parameter is inconsistent with the parameter of another. For example, the parameter “A” may only be set if parameter “B” is set to true.	Change the value of one of the parameters described in the error message in accordance with the instructions.
InvalidCurrencyException	The provided currency code is not valid.	Refer to the error message for valid currency codes.
InvalidParameterFormatException	A parameter was not provided in the correct format. The name of the format is included in the error message.	Modify the parameter value to conform to the format.
InvalidParameterValueException	An invalid value was provided for a parameter.	Change the value of the parameter to one of the valid values mentioned in the error message.
InvalidSortByCriteriaException	Invalid sort criteria have been provided.	Refer to the error message for valid sort criteria.
InvalidSortOrderException	An invalid sort direction has been provided.	Change the value of the sort direction to either “Ascending” or “Descending”.
InvalidUserNameAndPasswordException	The provided user name and/or password to the function were incorrect.	Make sure that the user name and password are correct.
ListCountDoesNotEqualCountParameterException	The number of items in a list parameter does not equal the value provided in a parameter that represents the count of said items.	Change the value of the count parameter to match the size of the provided list.
ListSizeExceedsMaximumException	The number of items in a list parameter has exceeded the maximum allowed number.	Reduce the number of items in the list to the allowed number of items.

MaxValueLessThanMinValueException	The value of a parameter representing a maximum value is less than the value of a parameter representing a minimum value. The names and values of the parameters involved are described in the error message.	Change the value of the maximum value parameter to be greater than the value of the minimum value parameter, or vice versa.
MissingAlternativeParametersException	Occurs when no parameters in a group of alternative parameters have been entered into a function.	Make sure that one of the parameters mentioned in the error message has been entered.
MissingRequiredParameterException	A required parameter was not provided to the function. The name of the required parameter is included in the error message.	Provide a value for the required parameter.
ParameterGroupIncompleteException	Not all parameters have been provided in a group of parameters that must be entered together.	Provide values for all or none of the parameters described in the error message.
ParameterOutOfRangeException	The value of a parameter was out of range.	Change the value of the parameter to fall within the range described in the message.
ParameterValueGreaterThanMaximumException	The value of a parameter is greater than the allowed maximum value.	Change the value of the parameter to be equal to or less than the described maximum value.
ParameterValueLessThanMinimumException	The value of a parameter is less than the allowed minimum value.	Change the value of the parameter to be equal to or greater than the described minimum value.
ToDatePrecedesFromDateException	The value of a parameter representing a "to" date (e.g. "checkOutDate") precedes a corresponding date parameter representing a "from" date (e.g. "checkInDate")	Change the value of the "to" date parameter to a later date than the value of the "from" date parameter.
TooManyAlternativeParametersException	Occurs when too many parameters in a group of alternative parameters have been entered into a function.	Make sure that the correct number of parameters (as mentioned in the error message) has been entered.

Searching

This function is used to search for available rooms/apartments/villas in the system on a given date.

NOTE!

A new version SearchV2 replaces the old Search method, including all its parameters and the specific ones available through MarketSearch endpoint, i.e. *customerCountry* and *B2C*, that will allow to dynamically advise client nationality and point of sale type.

Non-static searching

Example URL with input for non-static results:

```
http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/SearchV2?userName=[YourUsername]&password=[YourPassword]&language=en&currencies=EUR&checkInDate=2017-12-02&checkOutDate=2017-12-04&numberOfRooms=1&destination=&destinationID=695&hotelIDs=&resortIDs=&accommodationTypes=&numberOfAdults=2&numberOfChildren=0&childrenAges=&infant=0&sortBy=&sortOrder=&exactDestinationMatch=&blockSuperdeal=&showTransfer=&mealIDs=&showCoordinates=&showReviews=&referencePointLatitude=&referencePointLongitude=&maxDistanceFromReferencePoint=&minStarRating=&maxStarRating=&featureIDs=&minPrice=&maxPrice=&themeIDs=&excludeSharedRooms=&excludeSharedFacilities=&prioritizedHotelIDs=&totalRoomsInBatch=&paymentMethodId=&CustomerCountry=gb&B2C=&showRoomTypeName=
```

Static searching

Example URL with input for static results:

```
http://xml.sunhotels.net/15/PostGet/StaticXMLAPI.aspx/SearchV2?userName=[YourUsername]&password=[YourPassword]&language=en&currencies=EUR&checkInDate=2017-12-02&checkOutDate=2017-12-04&numberOfRooms=1&destination=&destinationID=695&hotelIDs=&resortIDs=&accommodationTypes=&numberOfAdults=2&numberOfChildren=0&childrenAges=&infant=0&sortBy=&sortOrder=&exactDestinationMatch=&blockSuperdeal=&showTransfer=&mealIDs=&showCoordinates=&showReviews=&referencePointLatitude=&referencePointLongitude=&maxDistanceFromReferencePoint=&minStarRating=&maxStarRating=&featureIDs=&minPrice=&maxPrice=&themeIDs=&excludeSharedRooms=&excludeSharedFacilities=&prioritizedHotelIDs=&totalRoomsInBatch=&paymentMethodId=&CustomerCountry=gb&B2C=
```

Function inputs

The search function is to be called with the following parameters.

*: Required input

**: Required alternative (one, and only one, of these parameters must be specified)

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent performing the search.	Non-empty strings
* <i>password</i>	The password of the agent performing the search.	Non-empty strings
* <i>language</i>	The language that the search results will be displayed in.	A valid ISO code from the "GetLanguages" function.
* <i>currencies</i>	The currencies that the prices in the search results will be converted to.	String representing one or more currency codes, e.g. "GBP" or "GBP, USD"

<i>*checkInDate</i>	<p>This value determines the date when the client wishes to check into the hotel. Only the rooms that are available on this date are returned in the search results. This date must be a future date.</p> <p>Check-in date is set to DateTime format and time is set to 00:00:000 as it is referring to day, not time.</p> <p>earliestNonFreeCancellationDate tag is set only in Book response, but can be calculated with the hotel time zone (GetStaticHotelsAndRooms) and Check-in time (local time 12:00 pm on Check-in Date)</p>	<p>A date in the format 'YYYY-MM-DD'</p> <p>YYYY = Year</p> <p>MM = Month number</p> <p>DD = Day number</p>
<i>*checkOutDate</i>	<p>This value determines the date when the client wishes to check out of the hotel. Only the rooms that are available on this date are returned in the search results. This date cannot be earlier than <i>checkInDate</i> and must be a future date.</p>	<p>A date in the format 'YYYY-MM-DD', where</p> <p>YYYY = Year</p> <p>MM = Month number</p> <p>DD = Day number</p>
<i>*numberOfRooms</i>	The number of rooms that the client is interested in.	<p>An integer value between 1 and the number of adults specified in</p> <p><i>"numberOfAdults"</i></p>
<i>**destination</i>	The IATA-code for the airport that the client wishes to be available for transfer to or from the room(s).	String value representing an IATA-code
<i>**destinationID</i>	The ID of the destination where the client wishes to look for rooms.	An integer value referring to an existing destination.
<i>**hotelIDs</i>	The ID(s) of the hotel(s) where the client wishes to look for rooms.	<p>A string containing hotel IDs, separated by commas.</p> <p>Example:</p> <p>"12,34,56"</p>
<i>**resortIDs</i>	The ID(s) of the resort(s) where the client wishes to look for rooms.	<p>A string containing resort IDs, separated by commas.</p> <p>Example:</p> <p>"12,34,56"</p>
<i>accommodationTypes</i>	The types of accommodation the client wants to search for (separated by commas). If this parameter is not entered, all types of accommodations will be searched.	"Hotel", "Apartment", "Villa"
<i>*numberOfAdults</i>	The number of adults the client wants the rooms to be able to accommodate.	An integer value between 1 and 9.
<i>*numberOfChildren</i>	<p>The number of children the client wants the rooms to be able to accommodate.</p> <p>Note: Child is between 2 and 17 years old. 0-1 years old is infant, above 17 years is adult.</p>	An integer value between 0 and 9

<i>*childrenAges</i>	Ages of the children informed in <i>numberOfChildren</i> field. Allows values between 2 and 17	A string containing children ages, between 2 and 17, separated by commas. Example: "7,12"
<i>*infant</i>	This value determines if infants (Children 0-1 year old) are included in search. Note: MAX VALUE is 1 infant per searched/booked unit. All values bigger than 1 is handled as 1.	An integer value, 0 or 1.
<i>sortBy</i>	This value determines which property the search results should be sorted after. If this value is not specified, the results will be sorted by hotel names by default.	"Hotel", "Resort", "Price" or "Classification"
<i>sortOrder</i>	This value determines the sorting order of the search results. If this value is not specified, the results will be sorted in ascending order by default.	"Ascending" or "Descending"
<i>exactDestinationMatch</i>	When this value is set to "true" or "1" and the destination code is set, the search results will only contain rooms where the destination code is the primary destination code.	"true"/"1" or "false"/"0".
<i>blockSuperdeal</i>	When this value is set to "true" or "1", all rooms that are SUPERDEAL will be excluded from the search.	"true"/"1" or "false"/"0"
<i>mealIds</i>	The IDs of the meals that should be available on all rooms (separated by commas). Note: To retrieve the IDs to use for this parameter, use the new "GetMeals" function.	A string containing meal IDs, separated by commas. Example: "1,3,4,5,6"
<i>showCoordinates</i>	When this value is set to "true" or "1" and the agent is allowed to view the coordinates (in latitude and longitude) for a hotel, the coordinates are included in the search results. Note: The coordinates will only be included when performing a static search.	"true"/"1" to include coordinates.
<i>showReviews</i>	Determines whether average review values (rating, price value) should be displayed for each found hotel (where reviews have been made).	"true"/"1" to display review values.
<i>referencePointLatitude</i>	The latitude of the geographical point to use in order to only display hotels whose distance (in kilometers) from the geographical point are less than the value specified in the <i>maxDistanceFromReferencePoint</i> (see below). Note: If provided, the <i>referencePointLongitude</i> and <i>maxDistanceFromReferencePoint</i> parameters must also be provided.	A decimal value >= -90 and <= 90

<i>referencePointLongitude</i>	<p>The longitude of the geographical point to use in order to only display hotels whose distance (in kilometers) from the geographical point are less than the value specified in the <i>maxDistanceFromReferencePoint</i> (see below).</p> <p>Note: If provided, the <i>referencePointLatitude</i> and <i>maxDistanceFromReferencePoint</i> parameters must also be provided.</p>	A decimal value >= -180 and ≤ 180
<i>maxDistanceFromReferencePoint</i>	<p>The maximum distance (in kilometers) that all hotels should be located from the referencepoint (whose latitude is provided in <i>referencePointLatitude</i> and longitude is provided in <i>referencePointLongitude</i>)</p> <p>Note: If provided, the <i>referencePointLatitude</i> and <i>referencePointLongitude</i> parameters must also be provided.</p>	A decimal value ≥ 0
<i>minStarRating</i>	The minimum star rating that all hotels should have. Cannot be greater than the value provided in <i>maxStarRating</i> .	An integer value ≥ 1 and ≤ 5
<i>maxStarRating</i>	The maximum star rating that all hotels should have. Cannot be lesser than the value provided in <i>minStarRating</i> .	An integer value ≥ 1 and ≤ 5
<i>featureIds</i>	<p>The IDs of the features that the hotels and rooms should have.</p> <p>Note: To retrieve the IDs to use for this parameter, use the new “GetFeatures” function.</p>	A string containing feature IDs, separated by commas. Example: “1,2,10”
<i>minPrice</i>	The minimum total price (in the currency specified in the <i>currency</i> parameter) that the rooms may cost.	A decimal value ≥ 0
<i>maxPrice</i>	The maximum total price (in the currency specified in the <i>currency</i> parameter) that the rooms may cost.	A decimal value ≥ 0
<i>themeIds</i>	<p>The IDs of the themes that the hotels must have.</p> <p>Note: To get available theme IDs, use the “GetThemes” function.</p>	A string containing the theme IDs, separated by commas, e.g. “1,2”
<i>excludeSharedRooms</i>	Whether or not to exclude shared rooms. Note: When a room is shared the minimum age to book this room is 18.	“true”/“1” or “false”/“0”.
<i>excludeSharedFacilities</i>	Whether or not to exclude shared facilities. Note: When facilities are shared the minimum age to book this room is 18.	“true”/“1” or “false”/“0”.
<i>prioritizedHotelIds</i>	The ID of a hotel that should be prioritized in the search results.	A string containing the ID of the hotel to prioritize.

<i>totalRoomsInBatch</i>	A value representing total number of rooms requested by your client. Use this when 2 or more searches are made separately for same client.	An integer value between 1 and 9.
<i>paymentMethodId</i>	The payment method id the result must have.	An integer of either 1 or 2, where 1 = "Regular" and 2 = "Pay At Hotel". NULL will return both "Regular" and "Pay At Hotel".
<i>customerCountry</i>	A 2 letter country code representing the nationality of the client.	A valid country code from "GetDestinations" function or "downloadexport" xmltype = places.
<i>B2C</i>	Whether or not the client derives from a B2C/non-package point of sales.	"true"/"1" or "false"/"0".
<i>showRoomTypeName</i>	Determines whether the name for each room type should be displayed in the search results.	"true"/"1" to display room type name.

Function output

Depending on the web service that is used, the search function that is called will yield different outputs.

“NonStaticXMLAPI” web service

This is the web service that is used for only retrieving non-static information about hotels, etc. and ignoring static information such as the hotel name, address, etc.

Specification of output of search method can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx?op=SearchV2>

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<searchresult>
  <!--All retrieved hotels-->
  <hotels>
    <hotel isPrioritized="false">
      <!--The unique ID of the hotel-->
      <hotel.id></hotel.id>
      <!--The ID of the destination where the hotel is located-->
      <destination_id></destination_id>
      <!--The ID of the resort where the hotel is located-->
      <resort_id></resort_id>
      <!--Integer value indicating whether it is at all possible to
      add transfer to the hotel. NOTE! Availability of transfers to
      the hotel is depending on date and time for arrival and departure.
      To verify transfers available a TransferSearch has to be performed.
      1 = adding transfer possible, 0 = adding transfer not possible-->
      <transfer></transfer>
      <!--All room types that are available for the hotel.-->
      <roomtypes>
        <roomtype>
          <!--The ID of the room type.-->
          <roomtype.ID></roomtype.ID>
          <!--The Name of the room type.-->
          <roomtype.Name></roomtype.Name>
          <!--All rooms for the room type that belong
          to the hotel and match the search criteria-->
          <rooms>
            <room>
              <!--The ID of the room-->
              <id></id>
              <!--The number of bed spaces that are available for the room-->
              <beds></beds>
              <!--The number of extra bed spaces that are available for the room-->
              <extrabeds></extrabeds>
              <!--Meals that are available for the current room-->
              <meals>
                <meal>
                  <!--The ID of the meal.-->
                  <id></id>
                  <!--The ID of the meal label for this hotel.-->
                  <labelId></labelId>
                  <!--The combined prices for the room and the meal,
                  in all searched currencies-->
                  <prices>
                    <!--currency: The currency code that the price applies to
                    paymentMethods: The IDs of the payment methods that the
                    price applies to
                    (separated by whitespaces) -->
                    <price currency="" paymentMethods="">
                      <!--The amount is stored as content of the
```

```

        "price" element -->
        </price>
    </prices>
    <!--The discount that has been deducted from the price.-->
    <discount>
        <!--The ID of the discount type-->
        <typeId></typeId>
        <!--The discount amounts, in all searched currencies-->
        <amounts>
            <!--currency: The currency code that the discount
            amount applies to
            paymentMethods: The IDs of the payment methods that
            the discount amount applies to
            (separated by a single whitespace) -->
            <amount currency="" paymentMethods="">
                <!--The discount amount is stored as
                content of the "amount" element -->
            </amount>
        </amounts>
    </discount>
</meal>
</meals>
<!--The cancellation policies for the current
room during the specified stay-->
<cancellation_policies>
    <cancellation_policy>
        <!--The deadline for the cancellation policy, i.e. the
        number of hours before the check-in date after which
        you will be charged with a certain penalty when
        cancelling the booking. Empty deadline means that the
        policy is effective at the time of the booking.
        Examples: empty deadline and percentage 100 = no refund.
        deadline 0 = no show. -->
        <deadline></deadline>
        <!--The percentage of the total booking
        price that will be charged when the
        deadline of this policy is reached.-->
        <percentage></percentage>
    </cancellation_policy>
</cancellation_policies>
<!--Special notes regarding this room at a specific time.
NOTE: Important to inform client through booking process.-->
<notes>
    <!--start_date: The start date of the note,
    end_date: The end date of the note-->
    <note start_date="" end_date="">
        <!--The text content of the note-->
        <text></text>
    </note>
</notes>
<!--Determines whether the room is a "Super Deal" room-->
<isSuperDeal></isSuperDeal>
<!--Determines whether the room is a "Best Buy" room-->
<isBestBuy></isBestBuy>
<!--The available payment methods for the room-->
<paymentMethods>
    <!--id: The ID of the payment method. 1: Regular, 2: Pay at hotel
    For more information about payment methods, visit
    http://xml.sunhotels.net/15/PaymentMethods/ -->
    <paymentMethod id="">

```



```
        <!--Depending on the payment method,
        different properties may appear here.-->
        <property key="" value=""/>
        <property key="" value=""/>
    </paymentMethod>
</paymentMethods>
</room>
</rooms>
</roomtype>
</roomtypes>
<!--Contains review information for this hotel-->
<review>
    <!--Average rating (1 - 7)-->
    <rating></rating>
    <!--Average price value (1 - 7)-->
    <pricevalue></pricevalue>
    <!--Number of reviews-->
    <count></count>
    <!--Number of recommendations-->
    <recommendCount></recommendCount>
</review>
<notes />
<!--The distance between the hotel and the reference point-->
<distance></distance>
<codes>
    <!-- Codes.-->
    <code type="" value=""/></code>
</codes>
</hotel>
</hotels>
<!--This element only appears when an error has occurred.-->
<Error>
    <!--The type of error (exception).-->
    <ErrorType></ErrorType>
    <!--A description of the error.-->
    <Message></Message>
</Error>
</searchresult>
```

“StaticXMLAPI” web service

This is the web service that is used for including static information such as the hotel name, address, etc. in addition to the non-static information in the search results.

Specification of output of search method can be found at:

<http://xml.sunhotels.net/15/PostGet/StaticXMLAPI.asmx?op=SearchV2>

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<searchresult>
  <hotels>
    <hotel xsi:type="StaticHotel" isPrioritized="false">
      <!--The unique ID of the hotel-->
      <hotel.id></hotel.id>
      <!--The ID of the destination where the hotel is located-->
      <destination_id></destination_id>
      <!--The ID of the resort where the hotel is located-->
      <resort_id></resort_id>
      <!--Integer value indicating whether it is at all possible to
      add transfer to the hotel. NOTE! Availability of transfers to
      the hotel is depending on date and time for arrival and departure.
      To verify transfers available a TransferSearch has to be performed.
      1 = adding transfer possible, 0 = adding transfer not possible-->
      <transfer></transfer>
      <!--All room types that are available for the hotel.-->
      <roomtypes>
        <roomtype xsi:type="StaticRoomTypeWithRooms">
          <!--The ID of the room type.-->
          <roomtype.ID></roomtype.ID>
          <!--All rooms for the room type that belong
          to the hotel and match the search criteria-->
          <rooms>
            <room xsi:type="StaticRoom">
              <!--The ID of the room-->
              <id></id>
              <!--The number of bed spaces that are available for the room-->
              <beds></beds>
              <!--The number of extra bed spaces that are available for the room-->
              <extrabeds></extrabeds>
              <!--Meals that are available for the current room-->
              <meals>
                <meal xsi:type="StaticRoomMeal">
                  <!--The ID of the meal.-->
                  <id></id>
                  <!--The ID of the meal label for this hotel.-->
                  <labelId></labelId>
                  <!--The combined prices for the room and the meal,
                  in all searched currencies-->
                  <prices>
                    <!--currency: The currency code that the price applies to
                    paymentMethods: The IDs of the payment methods that the
                    price applies to
                    (separated by a whitespace) -->
                    <price currency="" paymentMethods="">
                      <!--The amount is stored as content of the
                      "price" element -->
                      </price>
                    </prices>
                    <!--The discount that is included in the price.-->
```

```

<discount>
  <!--The ID of the discount type-->
  <typeId></typeId>
  <!--The discount amounts, in all searched currencies-->
  <amounts>
    <!--currency: The currency code that the discount
    amount applies to
    paymentMethods: The IDs of the payment methods that
    the discount amount applies to
    (separated by a single whitespace) -->
    <amount currency="" paymentMethods="">
      <!--The discount amount is stored as
      content of the "amount" element -->
    </amount>
  </amounts>
</discount>
<!--The name of the meal (translated to the language specified
for the search-->
<name></name>
<!--The text for the meal label (translated to the language
specified for the search-->
<labelText></labelText>
</meal>
</meals>
<!--The cancellation policies for the current
room during the specified stay-->
<cancellation_policies>
  <cancellation_policy>
    <!--The deadline for the cancellation policy, i.e. the
    number of hours before the check-in date after which
    you will be charged with a certain penalty when
    cancelling the booking. Empty deadline means that the
    policy is effective at the time of the booking.
    Examples: empty deadline and percentage 100 = no refund.
    deadline 0 = no show. -->
    <deadline></deadline>
    <!--The percentage of the total booking
    price that will be charged when the
    deadline of this policy is reached.-->
    <percentage></percentage>
    <!--An auto-generated summary of the details
    of the cancellation policy-->
    <text></text>
  </cancellation_policy>
</cancellation_policies>
<!--Special notes regarding this room at a specific time.
NOTE: Important to inform client through booking process.-->
<notes>
  <!--start_date: The start date of the note,
  end_date: The end date of the note-->
  <note start_date="" end_date="">
    <!--The text content of the note-->
    <text></text>
  </note>
</notes>
<!--Whether the room is a "Super Deal" room-->
<isSuperDeal></isSuperDeal>
<!--Whether the room is a "Best Buy" room-->
<isBestBuy></isBestBuy>
<!--The available payment methods for the room-->

```

```

        <paymentMethods>
            <!--id: The ID of the payment
method. 1: Regular, 2: Pay at hotel
            name: The translated name of the payment method.
            For more information about payment methods, visit
            http://xml.sunhotels.net/15/PaymentMethods/ -->
            <paymentMethod xsi:type="StaticPaymentMethod" id="" name="">
                <!--Depending on the payment method,
                different properties may appear here.-->
                <property key="" value=""/>
                <property key="" value=""/>
            </paymentMethod>
        </paymentMethods>
        <!--The features of the room-->
        <features>
            <!--id: The id of the feature, name: The name of the feature-->
            <feature id="" name="" />
        </features>
        <!--The themes of the room-->
        <themes>
            <!--
id: The id of the theme, name: The name of the theme, image: The image associated
with the theme-->
            <theme id="" name="" image="" />
        </themes>
    </room>
</rooms>
<!--The name of the room type (translated to the language
specified for the current search)-->
<room.type></room.type>
<!--Whether rooms are shared for this room type, if true minimum age is
18 years.-->
<sharedRoom></sharedRoom>
<!--Whether facilities are shared for this room type, if true minimum
age is 18 years.-->
<sharedFacilities></sharedFacilities>
</roomtype>
</roomtypes>
<!--Contains review information for this hotel-->
<review>
    <!--Average rating (1 - 7)-->
    <rating></rating>
    <!--Average price value(1 - 7)-->
    <pricevalue></pricevalue>
    <!--Number of reviews-->
    <count></count>
    <!--Number of recommendations for the hotel-->
    <recommendCount></recommendCount>
</review>
<!--Special notes regarding this hotel at a specific time-->
<notes>
    <!--start_date: The start date of the note,
    end_date: The end date of the note-->
    <note start_date="" end_date="">
        <!--The text content of the note-->
        <text></text>
    </note>
</notes>

```

```

<!--Type of accommodation, e.g. "Hotel", "Apartment" or "Villa"-->
<type></type>
<!--The name of the accommodation-->
<name></name>
<!--First address line of the accommodation-->
<hotel.addr.1></hotel.addr.1>
<!--Second address line of the accommodation-->
<hotel.addr.2></hotel.addr.2>
<!--Zip code of the accommodation-->
<hotel.addr.zip></hotel.addr.zip>
<!--The city where the accommodation is located-->
<hotel.addr.city></hotel.addr.city>
<!--The state where the accommodation is located-->
<hotel.addr.state></hotel.addr.state>
<!--The country where the accommodation is located-->
<hotel.addr.country></hotel.addr.country>
<!--The country code of the country where the accommodation is located-->
<hotel.addr.countrycode></hotel.addr.countrycode>
<!--The complete address of the accommodation-->
<hotel.address></hotel.address>
<!--The URL to a map view of the accommodation-->
<hotel.mapurl></hotel.mapurl>
<!--The accomodation headline-->
<headline></headline>
<!--The description of the accommodation-->
<description></description>
<!--The name of the resort where the accommodation is located-->
<resort></resort>
<!--The name of the destination where the accommodation is located.-->
<destination></destination>
<!--The themes of the accommodation-->
<themes>
  <!--
id: The id of the theme, name: The name of the theme, image: The image associated
with the theme-->
    <theme id="" name="" image="" />
</themes>
<!--All images for the accommodation-->
<images>
  <!--The id of the image-->
  <image id="">
    <!--The full-size version of the image
    url: The url to the image
    (needs to be prepended with "https://hotelimages.sunhotels.net/HotelInfo
/hotelImage.aspx?id=10473272&full=1")
    height: The height of the image (in pixels)
    width: The width of the image (in pixels)-->
    <fullSizeImage url="" height="" width="" />
    <!--The small version of the image
    url: The url to the image
    (needs to be prepended with "https://hotelimages.sunhotels.net/HotelInfo
/hotelImage.aspx?id=10473272")
    height: The height of the image (in pixels)
    width: The width of the image (in pixels)-->
    <smallImage url="" height="" width="" />
  </image>
</images>
<!--The features of the accomodation-->
<features>

```

```

    <!--id: The id of the feature, name: The name of the feature-->
    <feature id="" name="" />
</features>
<!--The accommodation classification,
e.g. a digit between 1 and 5 plus an optional character, e.g. "+"-->
<classification></classification>
<!--
The coordinates for the hotel (if the "showCoordinates" parameter has been
set to "true"/"1" and the agent is allowed to view coordinates).-->
<coordinates>
    <!--The latitude coordinate for the hotel.-->
    <latitude></latitude>
    <!--The longitude coordinate for the hotel.-->
    <longitude></longitude>
</coordinates>
<!--A list of distance types linked to the hotel,
i.e. distances to various beaches, airports, etc.-->
<distanceTypes>
    <distanceType>
        <!--The ID of the distance type-->
        <hotelDistanceTypeId></hotelDistanceTypeId>
        <!--The description of the type of place the child distances refer to-->
        <description></description>
        <distances>
            <distance>
                <!--The distance (in meters)-->
                <distanceInMeters> </distanceInMeters>
                <!--The name of a place-->
                <placeName></placeName>
                <!--Coordinates for the described place-->
                <coordinates>
                    <!--The latitude coordinate-->
                    <latitude></latitude>
                    <!--The longitude coordinate-->
                    <longitude></longitude>
                </coordinates>
            </distance>
        </distances>
    </distanceType>
</distanceTypes>
<!--The time zone where the hotel is located-->
<timeZone></timeZone>
<!--Whether hotel is "Best Buy" or not.-->
<isBestBuy></isBestBuy>
<!--The distance between the hotel and the reference point-->
<distance></distance>
<codes>
    <!-- Codes.-->
    <code type="" value=""></code>
</codes>
</hotel>
</hotels>
<!--This element only appears when an error has occurred.-->
<Error>
    <!--The type of error (exception).-->
    <ErrorType></ErrorType>
    <!--A description of the error.-->
    <Message></Message>

```

```
</Error>
</searchresult>
```

Errors

Listed below are common errors that may appear under the “Error” element.
Should any errors occur that are not listed here, contact Sunhotels.

Error Type	Description	Resolution
InvalidEmailAddressException	The provided e-mail address was not valid.	Provide a valid e-mail address in the format “x@y.z”

Terminology

Listed below are common concepts that appear in the search and searchV2.

Concept	Description
SuperDeal	A room that is always non-Refundable and non-Amendable.
BestBuy	Informative tag about a room being a Best Buy.

Search V.3

The SearchV3 method includes all the parameters from the previous version but the response returns **Taxes** and **Fees**.

“NonStaticXMLAPI” web service

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<searchresult>
  <!--All retrieved hotels-->
  <hotels>
    <hotel isPrioritized="false">
      <!--The unique ID of the hotel-->
      <hotel.id></hotel.id>
      <!--The ID of the destination where the hotel is located-->
      <destination_id></destination_id>
      <!--The ID of the resort where the hotel is located-->
      <resort_id></resort_id>
      <!--Integer value indicating whether it is at all possible to
      add transfer to the hotel. NOTE! Availability of transfers to
      the hotel is depending on date and time for arrival and departure.
      To verify transfers available a TransferSearch has to be performed.
      1 = adding transfer possible, 0 = adding transfer not possible-->
      <transfer></transfer>
      <!--All room types that are available for the hotel.-->
      <roomtypes>
        <roomtype>
          <!--The ID of the room type.-->
          <roomtype.ID></roomtype.ID>
          <!--The Name of the room type.-->
          <roomtype.Name></roomtype.Name>
          <!--All rooms for the room type that belong
          to the hotel and match the search criteria-->
          <rooms>
            <room>
              <!--The ID of the room-->
              <id></id>
```

```

<!--The number of bed spaces that are available for the room-->
<beds></beds>
<!--The number of extra bed spaces that are available for the room-->
<extrabeds></extrabeds>
<!--Meals that are available for the current room-->
<meals>
  <meal>
    <!--The ID of the meal.-->
    <id></id>
    <!--The ID of the meal label for this hotel.-->
    <labelId></labelId>
    <!--The combined prices for the room and the meal,
    in all searched currencies-->
    <prices>
      <!--currency: The currency code that the price applies to
      paymentMethods: The IDs of the payment methods that the
      price applies to
      (separated by whitespaces) -->
      <price currency="" paymentMethods="">
        <!--The amount is stored as content of the "price" element -->
      </price>
    </prices>
    <!--The discount that has been deducted from the price.-->
    <discount>
      <!--The ID of the discount type-->
      <typeId></typeId>
      <!--The discount amounts, in all searched currencies-->
      <amounts>
        <!--currency: The currency code that the discount
        amount applies to
        paymentMethods: The IDs of the payment methods that
        the discount amount applies to
        (separated by a single whitespace) -->
        <amount currency="" paymentMethods="">
          <!--The discount amount is stored as
          content of the "amount" element -->
        </amount>
      </amounts>
    </discount>
    <!--The taxes for the room and the meal, in
    all searched currencies-->
    <taxes>
      <!--currency: The currency code that the price applies to-->
      <tax currency="">
        <!--The amount is stored as content of the "tax" element-->
      </tax>
    </taxes>
  </meal>
</meals>
<!--The cancellation policies for the current
room during the specified stay-->
<cancellation_policies>
  <cancellation_policy>
    <!--The deadline for the cancellation policy, i.e. the
    number of hours before the check-in date after which
    you will be charged with a certain penalty when
    cancelling the booking. Empty deadline means that the
    policy is effective at the time of the booking.
    Examples: empty deadline and percentage 100 = no refund.
    deadline 0 = no show. -->
    <deadline></deadline>
    <!--The percentage of the total booking
    price that will be charged when the
  
```



```

        deadline of this policy is reached.-->
        <percentage></percentage>
    </cancellation_policy>
</cancellation_policies>
<!--Special notes regarding this room at a specific time.
NOTE: Important to inform client through booking process.-->
<notes>
    <!--start_date: The start date of the note,
    end_date: The end date of the note-->
    <note start_date="" end_date="">
        <!--The text content of the note-->
        <text></text>
    </note>
</notes>
<!--Determines whether the room is a "Super Deal" room-->
<isSuperDeal></isSuperDeal>
<!--Determines whether the room is a "Best Buy" room-->
<isBestBuy></isBestBuy>
<!--The available payment methods for the room-->
<paymentMethods>
    <!--id: The ID of the payment method. 1: Regular, 2: Pay at hotel
    For more information about payment methods, visit
    http://xml.sunhotels.net/15/PaymentMethods/ -->
    <paymentMethod id="">
        <!--Depending on the payment method,
        different properties may appear here.-->
        <property key="" value=""/>
        <property key="" value=""/>
    </paymentMethod>
</paymentMethods>
<!--The fees for the current room-->
<fees>
    <fee>
        <!--The name of the fee-->
        <name></name>
        <!--The fee amounts, in all searched currencies-->
        <amounts>
            <!--currency: The currency code that the fee amount
            applies to -->
            <amount currency="">
                <!--The fee amount is stored as
                content of the "amount" element -->
                </amount>
            </amounts>
            <!--Determines whether the price includes the fee amount-->
            <includedInPrice></includedInPrice>
        </fee>
    </fees>
</room>
</rooms>
</roomtype>
</roomtypes>
<!--Contains review information for this hotel-->
<review>
    <!--Average rating (1 - 7)-->
    <rating></rating>
    <!--Average price value (1 - 7)-->
    <pricevalue></pricevalue>
    <!--Number of reviews-->
    <count></count>
    <!--Number of recommendations-->
    <recommendCount></recommendCount>

```

```

    </review>
    <notes />
    <!--The distance between the hotel and the reference point-->
    <distance></distance>
    <codes>
        <!-- Codes.-->
        <code type="" value=""></code>
    </codes>
</hotel>
</hotels>
<!--This element only appears when an error has occurred.-->
<Error>
    <!--The type of error (exception).-->
    <ErrorType></ErrorType>
    <!--A description of the error.-->
    <Message></Message>
</Error>
</searchresult>

```

“StaticXMLAPI” web service

XML output

```

<?xml version="1.0" encoding="utf-8"?>
<searchresult>
    <hotels>
        <hotel xsi:type="StaticHotel" isPrioritized="false">
            <!--The unique ID of the hotel-->
            <hotel.id></hotel.id>
            <!--The ID of the destination where the hotel is located-->
            <destination_id></destination_id>
            <!--The ID of the resort where the hotel is located-->
            <resort_id></resort_id>
            <!--Integer value indicating whether it is at all possible to
            add transfer to the hotel. NOTE! Availability of transfers to
            the hotel is depending on date and time for arrival and departure.
            To verify transfers available a TransferSearch has to be performed.
            1 = adding transfer possible, 0 = adding transfer not possible-->
            <transfer></transfer>
            <!--All room types that are available for the hotel.-->
            <roomtypes>
                <roomtype xsi:type="StaticRoomTypeWithRooms">
                    <!--The ID of the room type.-->
                    <roomtype.ID></roomtype.ID>
                    <!--All rooms for the room type that belong
                    to the hotel and match the search criteria-->
                    <rooms>
                        <room xsi:type="StaticRoom">
                            <!--The ID of the room-->
                            <id></id>
                            <!--The number of bed spaces that are available for the room-->
                            <beds></beds>
                            <!--The number of extra bed spaces that are available for the room-->
                            <extrabeds></extrabeds>
                            <!--Meals that are available for the current room-->
                            <meals>
                                <meal xsi:type="StaticRoomMeal">
                                    <!--The ID of the meal.-->
                                    <id></id>
                                    <!--The ID of the meal label for this hotel.-->
                                    <labelId></labelId>
                                </meal>
                            </meals>
                        </room>
                    </rooms>
                </roomtype>
            </roomtypes>
        </hotel>
    </hotels>

```

```

<!--The combined prices for the room and the meal,
in all searched currencies-->
<prices>
  <!--currency: The currency code that the price applies to
    paymentMethods: The IDs of the payment methods that the
    price applies to
    (separated by a whitespace) -->
  <price currency="" paymentMethods="">
    <!--The amount is stored as content of the
    "price" element -->
  </price>
</prices>
<!--The discount that is included in the price.-->
<discount>
  <!--The ID of the discount type-->
  <typeId></typeId>
  <!--The discount amounts, in all searched currencies-->
  <amounts>
    <!--currency: The currency code that the discount
    amount applies to
    paymentMethods: The IDs of the payment methods that
    the discount amount applies to
    (separated by a single whitespace) -->
    <amount currency="" paymentMethods="">
      <!--The discount amount is stored as
      content of the "amount" element -->
    </amount>
  </amounts>
</discount>
<!--The name of the meal (translated to the language specified
for the search-->
<name></name>
<!--The text for the meal label (translated to the language
specified for the search-->
<labelText></labelText>
<!--The taxes for the room and the meal, in
all searched currencies-->
<taxes>
  <!--currency: The currency code that the price applies to-->
  <tax currency="">
    <!--The amount is stored as content of the "tax" element-->
  </tax>
</taxes>
</meal>
</meals>
<!--The cancellation policies for the current
room during the specified stay-->
<cancellation_policies>
  <cancellation_policy>
    <!--The deadline for the cancellation policy, i.e. the
    number of hours before the check-in date after which
    you will be charged with a certain penalty when
    cancelling the booking. Empty deadline means that the
    policy is effective at the time of the booking.
    Examples: empty deadline and percentage 100 = no refund.
    deadline 0 = no show. -->
  <deadline></deadline>
  <!--The percentage of the total booking
  price that will be charged when the
  deadline of this policy is reached.-->
  <percentage></percentage>
  <!--An auto-generated summary of the details
  of the cancellation policy-->

```

```

        <text></text>
    </cancellation_policy>
</cancellation_policies>
<!--Special notes regarding this room at a specific time.
NOTE: Important to inform client through booking process.-->
<notes>
    <!--start_date: The start date of the note,
    end_date: The end date of the note-->
    <note start_date="" end_date="">
        <!--The text content of the note-->
        <text></text>
    </note>
</notes>
<!--Whether the room is a "Super Deal" room-->
<isSuperDeal></isSuperDeal>
<!--Whether the room is a "Best Buy" room-->
<isBestBuy></isBestBuy>
<!--The available payment methods for the room-->
<paymentMethods>
    <!--id: The ID of the payment
        method.
        1: Regular,
        2: Pay at hotel
        name: The translated name of the payment method.
        For more information about payment methods, visit
        http://xml.sunhotels.net/15/PaymentMethods/ -->
    <paymentMethod xsi:type="StaticPaymentMethod" id="" name="">
        <!--Depending on the payment method,
        different properties may appear here.-->
        <property key="" value=""/>
        <property key="" value=""/>
    </paymentMethod>
</paymentMethods>
<!--The features of the room-->
<features>
    <!--id: The id of the feature, name: The name of the feature-->
    <feature id="" name="" />
</features>
<!--The themes of the room-->
<themes>
    <!--
id: The id of the theme, name: The name of the theme, image: The image associated
with the theme-->
    <theme id="" name="" image="" />
</themes>
<!--The fees for the current room-->
<fees>
    <fee>
        <!--The name of the fee-->
        <name></name>
        <!--The fee amounts, in all searched currencies-->
        <amounts>
            <!--currency: The currency code that the fee amount
            applies to -->
            <amount currency="">
                <!--The fee amount is stored as
                content of the "amount" element -->
            </amount>
        </amounts>
        <!--Determines whether the price includes the fee amount-->
        <includedInPrice></includedInPrice>
    </fee>

```

```

        </fees>
    </room>
</rooms>
<!--The name of the room type (translated to the language
specified for the current search)-->
<room.type></room.type>
<!--Whether rooms are shared for this room type, if true minimum age is
18 years.-->
<sharedRoom></sharedRoom>
<!--Whether facilities are shared for this room type, if true minimum
age is 18 years.-->
<sharedFacilities></sharedFacilities>
</roomtype>
</roomtypes>
<!--Contains review information for this hotel-->
<review>
    <!--Average rating (1 - 7)-->
    <rating></rating>
    <!--Average price value (1 - 7)-->
    <pricevalue></pricevalue>
    <!--Number of reviews-->
    <count></count>
    <!--Number of recommendations for the hotel-->
    <recommendCount></recommendCount>
</review>
<!--Special notes regarding this hotel at a specific time-->
<notes>
    <!--start_date: The start date of the note,
        end_date: The end date of the note-->
    <note start_date="" end_date="">
        <!--The text content of the note-->
        <text></text>
    </note>
</notes>
<!--Type of accommodation, e.g. "Hotel", "Apartment" or "Villa"-->
<type></type>
<!--The name of the accommodation-->
<name></name>
<!--First address line of the accommodation-->
<hotel.addr.1></hotel.addr.1>
<!--Second address line of the accommodation-->
<hotel.addr.2></hotel.addr.2>
<!--Zip code of the accommodation-->
<hotel.addr.zip></hotel.addr.zip>
<!--The city where the accommodation is located-->
<hotel.addr.city></hotel.addr.city>
<!--The state where the accommodation is located-->
<hotel.addr.state></hotel.addr.state>
<!--The country where the accommodation is located-->
<hotel.addr.country></hotel.addr.country>
<!--The country code of the country where the accommodation is located-->
<hotel.addr.countrycode></hotel.addr.countrycode>
<!--The complete address of the accommodation-->
<hotel.address></hotel.address>
<!--The URL to a map view of the accommodation-->
<hotel.mapurl></hotel.mapurl>
<!--The accomodation headline-->
<headline></headline>
<!--The description of the accommodation-->
<description></description>
<!--The name of the resort where the accommodation is located-->
<resort></resort>

```

```

<!--The name of the destination where the accommodation is located.-->
<destination></destination>
<!--The themes of the accommodation-->
<themes>
  <!--
id: The id of the theme, name: The name of the theme, image: The image associated
with the theme-->
    <theme id="" name="" image="" />
</themes>
<!--All images for the accommodation-->
<images>
  <!--The id of the image-->
  <image id="">
    <!--The full-size version of the image
    url: The url to the image
    (needs to be prepended with "https://hotelimages.sunhotels.net/HotelInfo
/hotelImage.aspx?id=10473272&full=1")
    height: The height of the image (in pixels)
    width: The width of the image (in pixels)-->
    <fullSizeImage url="" height="" width="" />
    <!--The small version of the image
    url: The url to the image
    (needs to be prepended with "https://hotelimages.sunhotels.net/HotelInfo
/hotelImage.aspx?id=10473272")
    height: The height of the image (in pixels)
    width: The width of the image (in pixels)-->
    <smallImage url="" height="" width="" />
  </image>
</images>
<!--The features of the accomodation-->
<features>
  <!--id: The id of the feature, name: The name of the feature-->
  <feature id="" name="" />
</features>
<!--The accommodation classification,
e.g. a digit between 1 and 5 plus an optional character, e.g. "+"-->
<classification></classification>
<!--
The coordinates for the hotel (if the "showCoordinates" parameter has been
set to "true"/"1" and the agent is allowed to view coordinates).-->
<coordinates>
  <!--The latitude coordinate for the hotel.-->
  <latitude></latitude>
  <!--The longitude coordinate for the hotel.-->
  <longitude></longitude>
</coordinates>
<!--A list of distance types linked to the hotel,
i.e. distances to various beaches, airports, etc.-->
<distanceTypes>
  <distanceType>
    <!--The ID of the distance type-->
    <hotelDistanceTypeId></hotelDistanceTypeId>
    <!--The description of the type of place the child distances refer to-->
    <description></description>
    <distances>
      <distance>
        <!--The distance (in meters)-->
        <distanceInMeters> </distanceInMeters>
        <!--The name of a place-->
        <placeName></placeName>
        <!--Coordinates for the described place-->
        <coordinates>

```

```
        <!--The latitude coordinate-->
        <latitude></latitude>
        <!--The longitude coordinate-->
        <longitude></longitude>
    </coordinates>
</distance>
</distances>
</distanceType>
</distanceTypes>
<!--The time zone where the hotel is located-->
<timeZone></timeZone>
<!--Whether hotel is "Best Buy" or not.-->
<isBestBuy></isBestBuy>
<!--The distance between the hotel and the reference point-->
<distance></distance>
<codes>
    <!-- Codes.-->
    <code type="" value=""></code>
</codes>
</hotel>
</hotels>
<!--This element only appears when an error has occurred.-->
<Error>
    <!--The type of error (exception).-->
    <ErrorType></ErrorType>
    <!--A description of the error.-->
    <Message></Message>
</Error>
</searchresult>
```

PreBook

Note that a newer version of PreBook exist, see PreBook V2.

This method is to be used as step 2 in the context of a 3 step booking process, after the search and before the booking request. The purpose of the PreBook call is to validate the information retrieved in search. This will allow you to alert the final customer of any changes prior to completing the booking.

The PreBook response return **Notes** (Erratas, renovations, city taxes etc) important to share with the customer prior to proceeding with the final Book request.

Important: this method is only compatible with Search and Book methods.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/PreBook?userName=\[YourUsername\]&password=\[YourPassword\]¤cy=EUR&language=en&checkInDate=2017-11-23&checkOutDate=2017-11-25&roomId=33513&rooms=1&adults=1&children=1&childrenAges=5&infant=0&mealId=3&CustomerCountry=gb&B2C=0&searchPrice=100](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/PreBook?userName=[YourUsername]&password=[YourPassword]¤cy=EUR&language=en&checkInDate=2017-11-23&checkOutDate=2017-11-25&roomId=33513&rooms=1&adults=1&children=1&childrenAges=5&infant=0&mealId=3&CustomerCountry=gb&B2C=0&searchPrice=100)

Function inputs

*: Required input

**: Required only when *children* value is greater than 0

Parameter name	Description	Valid Input Values
* <i>username</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>currency</i>	The currency used for the booking price.	String representing a currency code, e.g. "GBP".
* <i>language</i>	This string will determine the language that the booking result will be displayed in.	A valid ISO code from the "GetLanguages" function.
* <i>checkInDate</i>	The date for the check-in.	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number
* <i>checkOutDate</i>	The date for the check-out.	A date in the format 'YYYY-MM-DD', where YYYY = Year MM = Month number DD = Day number

<i>*roomId</i>	The ID of the room to be booked.	Integer representing a room ID
<i>*rooms</i>	The number of rooms to be booked.	An integer value between 1 and the number of adult guests.
<i>*adults</i>	The number of adult guests.	An integer value > 0
<i>*children</i>	The number of child guests.	An integer value >= 0
<i>**childrenAges</i>	Ages of the children informed in <i>numberOfChildren</i> field. Allows values between 2 and 17	A string containing children ages, between 2 and 17, separated by commas. Example: "7,12"
<i>*infant</i>	This value determines if infants (Children 0-1 year old) are included in booking. Note: MAX VALUE is 1 infant per searched/booked unit. All values bigger than 1 is handled as 1.	An integer value, 0 or 1
<i>*mealId</i>	The ID of the meal that is to be booked.	A valid meal ID (can be fetched using the "GetMeals" function)
<i>customerCountry</i>	A 2 letter country code representing the nationality of the client. The same parameter as used in search are required.	A valid country code from "GetDestinations" function or "downloadexport" xmltype = places.
<i>B2C</i>	Whether or not the client derives from a B2C/non-package point of sales. The same parameter as used in search are required.	"true"/"1" or "false"/"0".
<i>searchPrice</i>	The price that was received in Search response for the selected roomId and mealId	Price, optional

Function outputs

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx?op=PreBook>

XML output

```
<?xml version="1.0" encoding="utf-8" ?>
<PreBookResult>
  <!-- -->
  <Notes>
    <!--List of errata, resort fee, city tax - important to display to client -->
    <Note end_date="" start_date="" >
      <Text></Text>
    </Note>
    <!--Operation code, to be used in Book request-->
    <PreBookCode></PreBookCode>
    <!--Definite price of the booking-->
```

```
<Price currency=""></Price>
<!--All cancellation policies that apply for this booking. -->
<CancellationPolicies>
  <CancellationPolicy>
    <!--The deadline for the cancellation policy, i.e. the
         number of hours before the check-in date after which
         you will be charged with a certain penalty when
         cancelling the booking. Empty deadline means that the
         policy is effective at the time of the booking.
         Examples: empty deadline and percentage 100 = no refund.
         deadline 0 = no show. -->
    <deadline></deadline>
    <!--The percentage of the total booking
         price that will be charged when the
         deadline of this policy is reached.-->
    <percentage></percentage>
    <!--An auto-generated summary of the details of
         the cancellation policy-->
    <text></text>
  </CancellationPolicy>
</CancellationPolicies>
<!--This element only appears when an error has occurred.-->
<Error>
  <!--The type of error (the exception type).-->
  <ErrorType></ErrorType>
  <!--A description of the error.-->
  <Message></Message>
</Error>
</PreBookResult>
```

PreBook V.2

This method is to be used as step 2 in the context of a 3 step booking process, after the search and before the booking request. The purpose of the PreBook call is to validate the information retrieved in search. This will allow you to alert the final customer of any changes prior to completing the booking.

The PreBook response return **Notes** (Erratas, renovations, city taxes etc) important to share with the customer prior to proceeding with the final Book request.

PreBook V2 also have an option to retrieve the price breakdown in the response.

Important: this method is only compatible with SearchV2 and BookV2 methods.

There are two ways to use this function:

- Including the specific roomId to be booked.
- Including the roomtypeld to be booked. In this case also HotelId and blockSuperDeal tag are mandatory.

Take in account that one method excludes the other, meaning you cannot combine a search using roomId with hotelId, roomtypeld nor blockSuperDeal.

Important

- This function is only compatible with SearchV2 and BookV2 methods.
- This function must only be used prior to a booking request, and must be used ones per booking.
- The PreBook code given at the response is valid for 30 minutes.

Example URL with input using the roomId:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/PreBookV2?userName=\[YourUsername\]&password=\[YourPassword\]¤cy=EUR&language=en&checkInDate=2017-11-23&checkOutDate=2017-11-25&roomId=33513&rooms=1&adults=1&children=1&childrenAges=5&infant=0&mealId=3&CustomerCountry=gb&B2C=0&searchPrice=100&showPriceBreakdown=true](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/PreBookV2?userName=[YourUsername]&password=[YourPassword]¤cy=EUR&language=en&checkInDate=2017-11-23&checkOutDate=2017-11-25&roomId=33513&rooms=1&adults=1&children=1&childrenAges=5&infant=0&mealId=3&CustomerCountry=gb&B2C=0&searchPrice=100&showPriceBreakdown=true)

Example URL with input using the roomtypeld:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/PreBookV2?userName=\[YourUsername\]&password=\[YourPassword\]¤cy=EUR&language=en&checkInDate=2017-11-23&checkOutDate=2017-11-25&hotelId=7588&roomtypeld=50&rooms=1&adults=1&children=1&childrenAges=5&infant=0&mealId=3&CustomerCountry=gb&B2C=0&searchPrice=100&blockSuperDeal=1&showPriceBreakdown=true](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/PreBookV2?userName=[YourUsername]&password=[YourPassword]¤cy=EUR&language=en&checkInDate=2017-11-23&checkOutDate=2017-11-25&hotelId=7588&roomtypeld=50&rooms=1&adults=1&children=1&childrenAges=5&infant=0&mealId=3&CustomerCountry=gb&B2C=0&searchPrice=100&blockSuperDeal=1&showPriceBreakdown=true)

Function inputs

*: Required input

**: Required only when *children* value is greater than 0

***: Required input if using *roomId*

****: Required input if using *roomtypeld*

Parameter name	Description	Valid Input Values
* <i>username</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings

<i>*currency</i>	The currency used for the booking price.	String representing a currency code, e.g. "GBP".
<i>*language</i>	This string will determine the language that the booking result will be displayed in.	A valid ISO code from the "GetLanguages" function.
<i>*checkInDate</i>	The date for the check-in. Check-in date is set to DateTime format And time is set to 00:00:000 as it is referring to day, not time. earliestNonFreeCancellationDate tag is set only in Book response, but can be calculated with the hotel time zone (GetStaticHotelsAndRooms) and Check-in time (local time 12:00 pm on Check-in Date)	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number
<i>*checkOutDate</i>	The date for the check-out.	A date in the format 'YYYY-MM-DD', where YYYY = Year MM = Month number DD = Day number
<i>***roomId</i>	The ID of the room to be booked. Excludes the hotelID & roomtypeID use.	Integer representing a room ID
<i>****hotelId</i>	The ID of the Hotel to be booked. Must be used with the roomtypeID & blockSuperDeal tag. Excludes the roomID use.	Integer representing a Hotel ID
<i>****roomtypeId</i>	The ID of the Room type to be booked. Must be used with the hotelId & blockSuperDeal tag. Excludes the roomID use.	Integer representing a Room type
<i>*rooms</i>	The number of rooms to be booked.	An integer value between 1 and the number of adult guests.
<i>*adults</i>	The number of adult guests.	An integer value > 0
<i>*children</i>	The number of child guests.	An integer value >= 0
<i>**childrenAges</i>	Ages of the children informed in <i>numberOfChildren</i> field. Allows values between 2 and 17	A string containing children ages, between 2 and 17, separated by commas. Example: "7,12"
<i>*infant</i>	This value determines if infants (Children 0-1 year old) are included in booking. Note: MAX VALUE is 1 infant per searched/booked unit. All values bigger than 1 is handled as 1.	An integer value, 0 or 1

<i>*mealId</i>	The ID of the meal that is to be booked.	A valid meal ID (can be fetched using the "GetMeals" function)
<i>customerCountry</i>	A 2 letter country code representing the nationality of the client. The same parameter as used in search are required.	A valid country code from "GetDestinations" function or "downloadexport" xmltype = places.
<i>B2C</i>	Whether or not the client derives from a B2C/non-package point of sales. The same parameter as used in search are required.	"true"/"1" or "false"/"0".
<i>searchPrice</i>	The price that was received in Search response for the selected roomId and mealId	Price, optional
<i>showPriceBreakdown</i>	When this value is set to "true" or "1" the response will show the price breakdown of the total price per guests, night and discounts.	"true"/"1" or "false"/"0".
<i>****blockSuperDeal</i>	When this value is set to "true" or "1", all rooms that are SuperDeal will be excluded from the prebook. Mandatory if searching by <i>roomtypeId</i> .	"true"/"1" or "false"/"0".

Function outputs

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx?op=PreBook>

XML output

```
<?xml version="1.0" encoding="utf-8" ?>
<PreBookV2Result>
  <!-- -->
  <Notes>
    <!--List of errata, resort fee, city tax - important to display to client -->
    <Note end_date="" start_date="" >
      <Text></Text>
    </Note>
    <RoomId> </RoomId>
    <!--Operation code, to be used in Book request-->
    <PreBookCode> </PreBookCode>
    <!--Definite currency and price of the booking-->
    <Price currency=""></Price>
    <!--This element only appears when value in showPriceBreakdown tag is
    true or 1. Price breakdown of the booking.
    Currency: The currency code that the price amount applies to.
    Total: Total price of the booking.
    From: First day of the stay period of the booking.
    To: Last day of the stay period of the booking. -->
    <PriceBreakdown currency="" total="" from="" to="">
      <!--Price breakdown per guest.
      Total: Total price of the booking per guest. This price
      includes the meals and discounts if applicable.
      Age: age of guest. Only appear in case the guest is a child-->
      <guest total="" age="">
```

```

        <!--Total price per guest and breakdown of that price per
        night for the base type.
        Type: Base price.
        Value: Value of the base price including meal. Discounts
        are not applied, where applicable, are not deducted
        from.
        Breakdown: price per night separated by '|'. Take in
        account that not all the nights would have the same
        value, but the sum of all of them will be equal to the
        value= -->
        <price type="base" value="" breakdown=""|"/>
        <!--Breakdown of the type price.
        Type: Value type (discount type name), where applicable
        Value: amount of type to apply on guest.-->
        <price type="" value=""/>
    </guest>
</PriceBreakdown>
<!--All cancellation policies that apply for this booking. -->
<!--Please keep in mind that default check-in time is considered 12:00hrs on
check-in date at hotel local time. Time difference calculations can be done
utilizing hotel local time in "GetStaticHotelsAndRooms" method. -->
<CancellationPolicies>
    <CancellationPolicy>
        <!--The deadline for the cancellation policy, i.e. the
        number of hours before the check-in date after which
        you will be charged with a certain penalty when
        cancelling the booking. Empty deadline means that the
        policy is effective at the time of the booking.
        Examples: empty deadline and percentage 100 = no refund.
        deadline 0 = no show. -->
        <deadline></deadline>
        <!-- The percentage of the total booking price that will be charged when
        the deadline of this policy is reached.-->
        <percentage></percentage>
        <!--An auto-generated summary of the details of
        the cancellation policy-->
        <text></text>
    </CancellationPolicy>
</CancellationPolicies>
<!--This element only appears when an error has occurred.-->
<Error>
    <!--The type of error (the exception type).-->
    <ErrorType></ErrorType>
    <!--A description of the error.-->
    <Message></Message>
</Error>
</PreBookV2Result>

```

PreBook V.3

The PreBookV3 method includes all the parameters from the previous version but the response returns **Taxes** and **Fees**.

Important: this method should be used with SearchV3 and BookV3 methods.

XML output

```
<?xml version="1.0" encoding="utf-8" ?>
<PreBookResult>
  <!-- -->
  <Notes>
    <!--List of errata, resort fee, city tax - important to display to client -->
    <Note end_date="" start_date="" >
      <Text></Text>
    </Note>
    <RoomId> </RoomId>
    <!--Operation code, to be used in Book request-->
    <PreBookCode> </PreBookCode>
    <!--Definite currency and price of the booking-->
    <Price currency=""></Price>
    <!--This element only appears when value in showPriceBreakdown tag istrue
    or 1. Price breakdown of the booking.
    Currency: The currency code that the price amount applies to.Total:
    Total price of the booking.
    From: First day of the stay period of the booking.To:
    Last day of the stay period of the booking. -->
    <PriceBreakdown currency="" total="" from="" to="">
      <!--Price breakdown per guest.
      Total: Total price of the booking per guest. This priceincludes the
      meals and discounts if applicable.
      Age: age of guest. Only appear in case the guest is a child-->
      <guest total="" age="">
        <!--Total price per guest and breakdown of that price pernight
        for the base type.
        Type: Base price.
        Value: Value of the base price including meal. Discountsare
        not applied, where applicable, are not deducted from.
        Breakdown: price per night separated by '|'. Take inaccount
        that not all the nights would have the same value, but the
        sum of all of them will be equal to thevalue= -->
        <price type="base" value="" breakdown=""|"/>
        <!--Breakdown of the type price.
        Type: Value type (discount type name), where applicableValue:
        amount of type to apply on guest.-->
        <price type="" value="">
      </guest>
    </PriceBreakdown>
    <!--All cancellation policies that apply for this booking. -->
    <!--Please keep in mind that default check-in time is considered 12:00hrs on
    check-in date at hotel local time. Time difference calculations can be done
    utilizing hotel local time in "GetStaticHotelsAndRooms" method. -->
    <CancellationPolicies>
      <CancellationPolicy>
        <!--The deadline for the cancellation policy, i.e. the
        number of hours before the check-in date after which
        you will be charged with a certain penalty when
        cancelling the booking. Empty deadline means that the
        policy is effective at the time of the booking.
        Examples: empty deadline and percentage 100 = no refund.
        deadline 0 = no show. -->
        <deadline></deadline>
      <!--
```

```

        The percentage of the total booking price that will be charged wh
        en the deadline of this policy is reached.-->
        <percentage></percentage>
    <!--An auto-generated summary of the details of
    the cancellation policy-->
    <text></text>
</CancellationPolicy>
</CancellationPolicies>
<!--Tax of the booking-->
<Tax currency=""></Tax>
<!--The fees of the booking-->
<Fees>
    <fee>
        <!--The name of the fee-->
        <name></name>
        <!--The amount and currency of the fee-->
        <amount currency=""></amount>
        <!--Determines whether the price includes the fee amount-->
        <includedInPrice></includedInPrice>
    </Fee>
</Fees>
<!--This element only appears when an error has occurred.-->
<Error>
    <!--The type of error (the exception type).-->
    <ErrorType></ErrorType>
    <!--A description of the error.-->
    <Message></Message>
</Error>
</PreBookResult>
```


Booking

This function is used to book a room/apartment/villa. Take in account that you can only use latin alphabet characters in the guests names (adult and children), <yourRef>, <invoiceRef> and <specialrequest> tags.

NOTE!

A new version BookV2 replaces the old Book method, including all its parameters and the specific ones available through MarketSearch endpoint, i.e. *customercountry* and *B2C*, that will allow to dynamically advise client nationality and point of sale type.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/BookV2?userName=\[YourUsername\]&password=\[YourPassword\]¤cy=EUR&language=en&email=noreply@sunhotels.net&checkInDate=2017-11-23&checkOutDate=2017-11-25&roomId=33513&rooms=1&adults=1&children=1&infant=0&yourRef=Test%20booking&specialrequest=&mealId=1&adultGuest1FirstName=Test&adultGuest1LastName=Adult&adultGuest2FirstName=&adultGuest2LastName=&adultGuest3FirstName=&adultGuest3LastName=&adultGuest4FirstName=&adultGuest4LastName=&adultGuest5FirstName=&adultGuest5LastName=&adultGuest6FirstName=&adultGuest6LastName=&adultGuest7FirstName=&adultGuest7LastName=&adultGuest8FirstName=&adultGuest8LastName=&adultGuest9FirstName=&adultGuest9LastName=&childrenGuest1FirstName=Test&childrenGuest1LastName=Child&childrenGuestAge1=8&childrenGuest2FirstName=&childrenGuest2LastName=&childrenGuestAge2=&childrenGuest3FirstName=&childrenGuest3LastName=&childrenGuestAge3=&childrenGuest4FirstName=&childrenGuest4LastName=&childrenGuestAge4=&childrenGuest5FirstName=&childrenGuest5LastName=&childrenGuestAge5=&childrenGuest6FirstName=&childrenGuest6LastName=&childrenGuestAge6=&childrenGuest7FirstName=&childrenGuest7LastName=&childrenGuestAge7=&childrenGuest8FirstName=&childrenGuest8LastName=&childrenGuestAge8=&childrenGuest9FirstName=&childrenGuest9LastName=&childrenGuestAge9=&customerEmail=&paymentMethodId=1&creditCardType=&creditCardNumber=&creditCardHolder=&creditCardCVV2=&creditCardExpYear=&creditCardExpMonth=&customerEmail=&invoiceRef=&CustomerCountry=gb&B2C=0](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/BookV2?userName=[YourUsername]&password=[YourPassword]¤cy=EUR&language=en&email=noreply@sunhotels.net&checkInDate=2017-11-23&checkOutDate=2017-11-25&roomId=33513&rooms=1&adults=1&children=1&infant=0&yourRef=Test%20booking&specialrequest=&mealId=1&adultGuest1FirstName=Test&adultGuest1LastName=Adult&adultGuest2FirstName=&adultGuest2LastName=&adultGuest3FirstName=&adultGuest3LastName=&adultGuest4FirstName=&adultGuest4LastName=&adultGuest5FirstName=&adultGuest5LastName=&adultGuest6FirstName=&adultGuest6LastName=&adultGuest7FirstName=&adultGuest7LastName=&adultGuest8FirstName=&adultGuest8LastName=&adultGuest9FirstName=&adultGuest9LastName=&childrenGuest1FirstName=Test&childrenGuest1LastName=Child&childrenGuestAge1=8&childrenGuest2FirstName=&childrenGuest2LastName=&childrenGuestAge2=&childrenGuest3FirstName=&childrenGuest3LastName=&childrenGuestAge3=&childrenGuest4FirstName=&childrenGuest4LastName=&childrenGuestAge4=&childrenGuest5FirstName=&childrenGuest5LastName=&childrenGuestAge5=&childrenGuest6FirstName=&childrenGuest6LastName=&childrenGuestAge6=&childrenGuest7FirstName=&childrenGuest7LastName=&childrenGuestAge7=&childrenGuest8FirstName=&childrenGuest8LastName=&childrenGuestAge8=&childrenGuest9FirstName=&childrenGuest9LastName=&childrenGuestAge9=&customerEmail=&paymentMethodId=1&creditCardType=&creditCardNumber=&creditCardHolder=&creditCardCVV2=&creditCardExpYear=&creditCardExpMonth=&customerEmail=&invoiceRef=&CustomerCountry=gb&B2C=0)

Function inputs

*: Required input

** Required input for adults and children included in booking

***: Required input when booking Pay at Hotel rooms (paymentMethodId = 2)

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>currency</i>	The currency used for the booking price.	String representing a currency code, e.g. "GBP".
* <i>language</i>	This string will determine the language that the booking result will be displayed in.	A valid ISO code from the "GetLanguages" function.
* <i>email</i>	The e-mail address of the agent. This is where the booking confirmation e-mail is sent.	String representing a valid e-mail address.
* <i>checkInDate</i>	The date for the check-in. Check-in date is set to DateTime format And time is set to 00:00:000 as it is referring to day, not time. earliestNonFreeCancellationDate tag is set only in Book response, but can be calculated with the hotel time zone (GetStaticHotelsAndRooms) and Check-in time (local time 12:00 pm on Check-in Date)	A date in the format "YYYY-MM-DD" YYYY = Year MM = Month number DD = Day number

<i>*checkOutDate</i>	The date for the check-out.	A date in the format 'YYYY-MM-DD', where YYYY = Year MM = Month number DD = Day number
<i>*roomId</i>	The ID of the room to be booked.	Integer representing a room ID
<i>*rooms</i>	The number of rooms to be booked.	An integer value between 1 and the number of adult guests.
<i>*adults</i>	The number of adult guests.	An integer value > 0
<i>*children</i>	The number of child guests.	An integer value >= 0
<i>*infant</i>	This value determines if infants (Children 0-1 year old) are included in booking. Note: MAX VALUE is 1 infant per searched/booked unit. All values bigger than 1 is handled as 1. A booking number or reference.	An integer value, 0 or 1
<i>yourRef</i>		A string representing your reference number.
<i>specialrequest</i>	Any special requests for the booking. Note: This field is only to be used for special requests, and should be entered in English.	A string containing the special requests.
<i>*mealId</i>	The ID of the meal that is to be booked.	A valid meal ID (can be fetched using the "GetMeals" function)
<i>**adultGuest1FirstName</i> <i>**adultGuest1LastName</i> <i>**adultGuest2FirstName</i> <i>**adultGuest2LastName</i> ... <i>**adultGuest8FirstName</i> <i>**adultGuest8LastName</i> <i>**adultGuest9FirstName</i> <i>**adultGuest9LastName</i>	The first and last names of the adult guest(s). Note: Using TBA may cause issues for the end customer in reaching the hotel. We recommend avoiding its use.	Strings representing first and last names, e.g. "John" and "Smith" adultGuest1FirstName & adultGuest1LastName must be the real holder of the booking. If the other names for the guest are not available must be filled with "TBA".
<i>**childrenGuest1FirstName</i> <i>**childrenGuest1LastName</i>	The first and last names of the first child guest.	Strings representing first and last names, e.g. "John" and "Smith"
<i>**childrenGuestAge1</i>	The age of the first child guest.	Integer where x >= 2 and x <= 17
<i>**childrenGuest2FirstName</i> <i>**childrenGuest2LastName</i>	The first and last names of the second child guest.	Strings representing first and last names, e.g. "John" and "Smith"
<i>**childrenGuestAge2</i>	The age of the second child guest.	Integer where x >= 2 and x <= 17
...		

**childrenGuest8FirstName **childrenGuest8LastName	The first and last names of the eighth child guest.	Strings representing first and last names, e.g. "John" and "Smith"
**childrenGuestAge8	The age of the eighth child guest.	Integer where $x \geq 2$ and $x \leq 17$
**childrenGuest9FirstName **childrenGuest9LastName	The first and last names of the ninth child guest.	Strings representing first and last names, e.g. "John" and "Smith"
**childrenGuestAge9	The age of the ninth child guest.	Integer where $x \geq 2$ and $x \leq 17$
paymentMethodId	The id of the payment method to be used.	An integer of either 1 or 2, where 1 = "Regular" and 2 = "Pay At Hotel"
***creditCardType	The credit card type	Either "VISA" or "MasterCard"
***creditCardNumber	The credit card number	String representing a credit card number
***creditCardHolder	The name of the credit card holder	String representing a name
***creditCardCVV2	The CCV2 code of the credit card	String representing a CVV2 code
***creditCardExpYear	The expiration year of the credit card	Integer representing a year
***creditCardExpMonth	The expiration month of the credit card	Integer where $x \geq 1$ and $x \leq 12$
customerEmail	Email address of the customer. Non-required field. Only to be used in separate agreement with Sunhotels.	String representing an email address.
invoiceRef	An invoice reference	A string representing your invoice reference
commissionAmountInHotelCurrency	Commission added by client for "pay at hotel" bookings. Not to be used without previous agreement with Sunhotels.	String representing a percentage.
customerCountry	A 2 letter country code representing the nationality of the client. The same parameter as used in search are required.	A valid country code from "GetDestinations" function or "downloadexport" xmltype = places.
B2C	Whether or not the client derives from a B2C/non-package point of sales. The same parameter as used in search are required.	"true"/"1" or "false"/"0".
PreBookCode	Code retrieved in PreBook response Mandatory if PreBook is used.	String.

Function outputs

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx?op=BookV2>

XML output

```
<?xml version="1.0" encoding="utf-8" ?>
<bookResult>
  <!--Represents the created booking-->
  <booking>
    <!--The ID of the booking-->
    <bookingnumber></bookingnumber>
    <!--The ID of the booked hotel-->
    <hotel.id></hotel.id>
    <!--The name of the booked hotel-->
    <hotel.name></hotel.name>
    <!--The address of the booked hotel.-->
    <hotel.address></hotel.address>
    <!--The phone number to the booked hotel.-->
    <hotel.phone></hotel.phone>
    <!--The number of booked rooms.-->
    <numberofrooms></numberofrooms>
    <!--The type of room that was booked (translated to the specified language)-->
    <room.type></room.type>
    <!--The type of room that was booked (in English).-->
    <room.englishType></room.englishType>
    <!--The id of the meal that was booked-->
    <mealId></mealId>
    <!--The meal that was booked (translated to the specified language)-->
    <meal></meal>
    <!--The meal label for booked meal.-->
    <mealLabel></mealLabel>
    <!--The meal that was booked (in English).-->
    <englishMeal></englishMeal>
    <!--The meal label for booked meal (in English).-->
    <englishMealLabel></englishMealLabel>
    <!--The check-in date for the booking-->
    <checkindate></checkindate>
    <!--The check-out date for the booking-->
    <checkoutdate></checkoutdate>
    <!--The prices for the booking, which are
    expressed in at least the booked currency
    as well as in Euros
    currency: The currency code that the price
    applies to
    paymentMethods: The payment methods the price
    applies to (always the booked payment method)-->
    <prices>
      <price currency="" paymentMethods=""></price>
      <price currency="" paymentMethods=""></price>
    </prices>
    <!--The currency of the booked price-->
    <currency></currency>
    <!--The date when the booking was made.-->
    <bookingdate></bookingdate>
    <!--The time zone in which the booking was made.-->
    <bookingdate.timezone></bookingdate.timezone>
    <!--All cancellation policies that apply for this booking.-->
    <cancellationpolicies>
      <!--The deadline for the cancellation policy, i.e. the
      number of hours before the check-in date after which
      you will be charged with a certain penalty when
      cancelling the booking. Empty deadline means that the
      policy is effective at the time of the booking.
      Example: empty deadline and percentage 100 = no refund
      deadline 0 = no show.-->
```

```

    <deadline></deadline>
    <!--The percentage of the total booking price that will
    be charged when the deadline of this policy is reached.-->
    <percentage></percentage>
    <!--An auto-generated summary of the details of
    the cancellation policy-->
    <text></text>
</cancellationpolicies>
<!--The earliest date when cancelling the booking will not be free
(in Central European Standard Time)-->
<earliestNonFreeCancellationDate.CET>
</earliestNonFreeCancellationDate.CET>
<!--The earliest date when cancelling the booking will not be free
(in the local time-zone of the destination where the hotel is located)-->
<earliestNonFreeCancellationDate.Local>
</earliestNonFreeCancellationDate.Local>
<!--The reference used when booking-->
<yourref></yourref>
<!--The URL to the voucher associated with the booking.-->
<voucher></voucher>
<!--Emergency contact details and additional important information
associated with the booking required to pass to the client-->
<bookedBy></bookedBy>
<!--An integer indicating whether a transfer has been booked
(1: yes, 0: no, 2: transfer has been cancelled)-->
<transferbooked></transferbooked>
<!--The booked payment method
    id: The id of the payment method. 1: Invoice/Pay now, 2: Pay at hotel
    name: The translated name of the payment method
    For more information about payment methods, visit
    http://xml.sunhotels.net/15/PaymentMethods/ -->
<paymentMethod id="" name="">
    <!--Depending on the payment method,
    different properties may appear here.-->
    <property key="" value=""/>
    <property key="" value=""/>
</paymentMethod>
<!--Special notes regarding the booked hotel
(translated to the specified language).
NOTE: Important to inform client through booking process.-->
<hotelNotes>
    <!--start_date: The start date of the note,
    end_date: The end date of the note-->
    <hotelNote start_date="" end_date="">
        <!--The text content of the note-->
        <text></text>
    </hotelNote>
</hotelNotes>
<!--Special notes regarding the booked hotel (in English)-->
<englishHotelNotes>
    <!--start_date: The start date of the note,
    end_date: The end date of the note-->
    <englishHotelNote start_date="" end_date="">
        <!--The text content of the note-->
        <text></text>
    </englishHotelNote>
</englishHotelNotes>
<!--Special notes regarding the booked room

```

```

(translated to the specified language)-->
<roomNotes>
  <!--start_date: The start date of the note,
    end_date: The end date of the note-->
  <roomNote start_date="" end_date="">
    <!--The text content of the note-->
    <text></text>
  </roomNote>
</roomNotes>
<!--Special notes regarding the booked room (in English)-->
<englishRoomNotes>
  <!--start_date: The start date of the note,
    end_date: The end date of the note-->
  <englishRoomNote start_date="" end_date="">
    <!--The text content of the note-->
    <text></text>
  </englishRoomNote>
</englishRoomNotes>
<!--The invoice reference used when booking-->
<invoiceref></invoiceref>
</booking>
<!--This element only appears when an error has occurred.-->
<Error>
  <!--The type of error (the exception type).-->
  <ErrorType></ErrorType>
  <!--A description of the error.-->
  <Message></Message>
</Error>
</bookResult>

```

Errors

Listed below are common errors that may appear under the “Error” element.
Should any errors occur that are not listed here, contact Sunhotels.

Error Type	Description	Resolution
InvalidEmailAddressException	The provided e-mail address was not valid.	Provide a valid e-mail address in the format “x@y.z”
AgentMustUseCreditCardBookingException	The agent performing the booking has indicated that he/she does not want to use credit card payment, but is required to do so.	Change the value of the “useCreditCardPayment” to “true”.
AgentMayNotUseCreditCardBookingException	The agent performing the booking has indicated that he/she wants to use credit card payment, but is not allowed to.	Change the value of the “useCreditCardPayment” to “false”.
RoomSuspendedException	The room that the agent is trying to book is suspended.	Enter the ID of a room that is not suspended.

InvalidMealForRoomException	The provided meal is not valid for the room that the agent is trying to book.	Enter the ID of a meal that is available for the room to book.
CreditCardPaymentException	The credit card used to pay for the booking was denied.	Make sure that the correct amount is available on the credit card, and that it has not expired.
EmptyLastNameForGuestNotAllowedException	The last name for a guest was empty, which is not allowed.	Change the value of the parameter mentioned in the error message to contain a non-empty string.
InvalidChildAgeFormatException	The age entered for a child guest was either not a valid integer OR outside the allowed range of 2 to 17 years.	Modify the provided value so that it is a valid integer between 2 and 17.
NoRoomAvailabilityException	The room that the agent is trying to book is not available.	Try to book another room.
ChildAgeOutOfRangeException	The age entered for a child guest is out of range.	Enter an age between 2 and 17.
AgentPaymentMethodInvalidException	The payment method specified is not valid for this agent.	Enter a valid payment method id of either 1 or 2.
CreditCardNotValidException	The provided credit card information is not valid.	Observe the error message for more information about the erroneous parameter.
PriceMismatchException	Final price did not match the price returned in the PreBook.	Final price was changed and new price is returned together with new PreBookCode. After confirming new price with client book again using new PreBookCode.
InvalidPrebookCode	Exceeded 30 min. From Prebook to Book sections.	Do not exceed more than 30 minutes between Sections.

Book V.3

The BookV3 method includes all the parameters from the previous version but the response returns **Taxes** and **Fees**.

XML output

```
<?xml version="1.0" encoding="utf-8" ?>
<bookResult>
  <!--Represents the created booking-->
  <booking>
    <!--The ID of the booking-->
    <bookingnumber></bookingnumber>
    <!--The ID of the booked hotel-->
    <hotel.id></hotel.id>
    <!--The name of the booked hotel-->
    <hotel.name></hotel.name>
    <!--The address of the booked hotel.-->
    <hotel.address></hotel.address>
    <!--The phone number to the booked hotel.-->
    <hotel.phone></hotel.phone>
    <!--The number of booked rooms.-->
    <numberofrooms></numberofrooms>
    <!--The type of room that was booked (translated to the specified language)-->
    <room.type></room.type>
    <!--The type of room that was booked (in English).-->
    <room.englishType></room.englishType>
    <!--The id of the meal that was booked-->
    <mealId></mealId>
    <!--The meal that was booked (translated to the specified language)-->
    <meal></meal>
    <!--The meal label for booked meal.-->
    <mealLabel></mealLabel>
    <!--The meal that was booked (in English).-->
    <englishMeal></englishMeal>
    <!--The meal label for booked meal (in English).-->
    <englishMealLabel></englishMealLabel>
    <!--The check-in date for the booking-->
    <checkindate></checkindate>
    <!--The check-out date for the booking-->
    <checkoutdate></checkoutdate>
    <!--The prices for the booking, which are
    expressed in at least the booked currency
    as well as in Euros
    currency: The currency code that the price
    applies to
    paymentMethods: The payment methods the price
    applies to (always the booked payment method)-->
    <prices>
      <price currency="" paymentMethods=""></price>
      <price currency="" paymentMethods=""></price>
    </prices>
    <!--The currency of the booked price-->
    <currency></currency>
    <!--The date when the booking was made.-->
    <bookingdate></bookingdate>
    <!--The time zone in which the booking was made.-->
    <bookingdate.timezone></bookingdate.timezone>
    <!--All cancellation policies that apply for this booking.-->
    <cancellationpolicies>
      <!--The deadline for the cancellation policy, i.e. the
      number of hours before the check-in date after which
```


you will be charged with a certain penalty when cancelling the booking. Empty deadline means that the policy is effective at the time of the booking.

Examples: empty deadline and percentage 100 = no refund. | deadline 0 = no show. -->

```

<deadline></deadline>
<!--The percentage of the total booking price that will
be charged when the deadline of this policy is reached.-->
<percentage></percentage>
<!--An auto-generated summary of the details of
the cancellation policy-->
<text></text>
</cancellationpolicies>
<!--The earliest date when cancelling the booking will not be free
(in Central European Standard Time)-->
<earliestNonFreeCancellationDate.CET>
</earliestNonFreeCancellationDate.CET>
<!--The earliest date when cancelling the booking will not be free
(in the local time-zone of the destination where the hotel is located)-->
<earliestNonFreeCancellationDate.Local>
</earliestNonFreeCancellationDate.Local>
<!--The reference used when booking-->
<yourref></yourref>
<!--The URL to the voucher associated with the booking.-->
<voucher></voucher>
<!--Emergency contact details and additional important information
associated with the booking required to pass to the client-->
<bookedBy></bookedBy>
<!--An integer indicating whether a transfer has been booked
(1: yes, 0: no, 2: transfer has been cancelled)-->
<transferbooked></transferbooked>
<!--The booked payment method
    id: The id of the payment method. 1: Invoice/Pay now, 2: Pay at hotel
    name: The translated name of the payment method
    For more information about payment methods, visit
    http://xml.sunhotels.net/15/PaymentMethods/ -->
<paymentMethod id="" name="">
    <!--Depending on the payment method,
    different properties may appear here.-->
    <property key="" value=""/>
    <property key="" value=""/>
</paymentMethod>
<!--Special notes regarding the booked hotel
(translated to the specified language).
NOTE: Important to inform client through booking process.-->
<hotelNotes>
    <!--start_date: The start date of the note,
    end_date: The end date of the note-->
    <hotelNote start_date="" end_date="">
        <!--The text content of the note-->
        <text></text>
    </hotelNote>
</hotelNotes>
<!--Special notes regarding the booked hotel (in English)-->
<englishHotelNotes>
    <!--start_date: The start date of the note,
    end_date: The end date of the note-->
    <englishHotelNote start_date="" end_date="">
        <!--The text content of the note-->
        <text></text>
    </englishHotelNote>
</englishHotelNotes>
<!--Special notes regarding the booked room

```

```

(translated to the specified language)-->
<roomNotes>
  <!--start_date: The start date of the note,
    end_date: The end date of the note-->
  <roomNote start_date="" end_date="">
    <!--The text content of the note-->
    <text></text>
  </roomNote>
</roomNotes>
<!--Special notes regarding the booked room (in English)-->
<englishRoomNotes>
  <!--start_date: The start date of the note,
    end_date: The end date of the note-->
  <englishRoomNote start_date="" end_date="">
    <!--The text content of the note-->
    <text></text>
  </englishRoomNote>
</englishRoomNotes>
<!--The invoice reference used when booking-->
<invoiceref></invoiceref>
<!--The taxes for the booking, which are
expressed in at least the booked currency
as well as in Euros
currency: The currency code that the tax
applies to-->
<taxes>
  <tax currency=""></tax>
  <tax currency=""></tax>
</taxes>
<!--The fees for the booking-->
<fees>
  <fee>
    <!--The name of the fee-->
    <name></name>
    <!--The fee amounts, which are
expressed in at least the booked
currency as well as in Euros-->
    <amounts>
      <!--currency: The currency code that the fee amount applies to
      -->
      <amount currency=""></amount>
      <amount currency=""></amount>
    </amounts>
    <!--Determines whether the price includes the fee amount-->
    <includedInPrice></includedInPrice>
  </fee>
</fees>
</booking>
<!--This element only appears when an error has occurred.-->
<Error>
  <!--The type of error (the exception type).-->
  <ErrorType></ErrorType>
  <!--A description of the error.-->
  <Message></Message>
</Error>
</bookResult>

```

Retrieving room type list

This function is used to retrieve a list of all room types and codes in the specified language.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetRoomTypes?userName=\[YourUsername\]&password=\[YourPassword\]&language=en](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetRoomTypes?userName=[YourUsername]&password=[YourPassword]&language=en)

Function inputs

*: Required input

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>language</i>	This string will determine the language that the room types will be displayed in.	A valid ISO code from the "GetLanguages" function.

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx?op=GetRoomTypes>

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<getRoomTypesResult>
  <!--All room types that were retrieved.-->
  <roomTypes>
    <roomType>
      <!--The unique ID of the room type.-->
      <id></id>
      <!--The name of the room type in the specified language.-->
      <name></name>
      <!--Whether rooms are shared for this room type, if true minimum age is 18
years. -->
      <!--0 = Not shared, 1 = Shared -->
      <sharedRoom></sharedRoom>
      <!--Whether facilities are shared for this room type, if true minimum age is 18
years. (1: yes, 0: no).-->
      <sharedFacilities></sharedFacilities>
    </roomType>
  </roomTypes>
  <!--This element only appears when an error has occurred.-->
  <Error>
    <!--The type of error (the exception type).-->
    <ErrorType></ErrorType>
    <!--A description of the error.-->
    <Message></Message>
  </Error>
</getRoomTypesResult>
```

Checking price difference for an amendment

This function is used to check if an amendment is possible, check the price difference that the amendment will cause and indicate what type of amendment that renders the best price. An amendment will either cancel the existing booking and create a new booking for the new stay (FullPeriod) or keep the old booking active and book the extra nights in a separate booking (ExtraDays).

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/AmendmentPriceRequest?userName=\[YourUsername\]&password=\[YourPassword\]&bookingID=1454020&checkInDate=2013-01-01&checkOutDate=2013-01-05&language=en](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/AmendmentPriceRequest?userName=[YourUsername]&password=[YourPassword]&bookingID=1454020&checkInDate=2013-01-01&checkOutDate=2013-01-05&language=en)

Function inputs

*: Required input

Parameter name	Description	Valid Input Values
*username	The name of the user that is performing the amendment operation.	Non-empty strings
*password	The password of the user that is performing the amendment operation.	Non-empty strings
*bookingID	The ID of the booking that is to be amended.	String containing numerical characters
*checkInDate	The date for the check-in after amendment.	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number
*checkOutDate	The date for the check-out after amendment.	A date in the format 'YYYY-MM-DD', where YYYY = Year MM = Month number DD = Day number
*language	Determines the language that the response will be displayed in.	A valid ISO code from the "GetLanguages" function.

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx?op=AmendmentPriceRequest>

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<amendmentPriceRequestResult xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://xml.sunhotels.net/15/">
  <!--The price increase or reduction of the requested amendment.
  Will always be in the currency in which the booking was created.-->
  <Price currency="EUR"></Price>
  <!--The room ID for the booking,
  can be different from the original booking room ID.-->
  <RoomId></RoomId>
  <!--Special hotel notes regarding this hotel at a specific time.
  NOTE: Important to inform client through booking process. -->
  <hotelnotes>
    <!--start_date: The start date of the note,
    end_date: The end date of the note-->
    <note start_date="" end_date="">
      <!--The text content of the note-->
      <text></text>
    </note>
  </hotelnotes>
  <!--Special room notes regarding this hotel at a specific time-->
```

```

<roomnotes>
  <!--start_date: The start date of the note,
  end_date: The end date of the note-->
  <note start_date="" end_date="">
    <!--The text content of the note-->
    <text></text>
  </note>
</roomnotes>
<!--The cancellation policies for the current room during the specified stay-->
<cancellation_policies>
  <cancellation_policy>
    <!--The deadline for the cancellation policy, i.e. the
    number of hours before the check-in date after which
    you will be charged with a certain penalty when
    cancelling the booking. Empty deadline means that the
    policy is effective at the time of the booking.
    Examples: empty deadline and percentage 100 = no refund.
    deadline 0 = no show. -->
    <deadline></deadline>
    <!--The percentage of the total booking
    price that will be charged when the
    deadline of this policy is reached.-->
    <percentage></percentage>
  </cancellation_policy>
</cancellation_policies>
<!--Enumeration of the type of amendment that can be done;
FullPeriod, ExtraDays or NoAmendmentAvailable.-->
< BookingType>FullPeriod</BookingType>
/amendmentPriceRequestResult

```

Errors

Listed below are common errors that may appear under the “Error” element.
Should any errors occur that are not listed here, contact Sunhotels.

Error Type	Description	Resolution
AmendmentNotPossibleException	The amendment requested could not be performed.	Check input parameters.

Booking an amendment

This function is used to book an amendment. It is only possible to amend nights to a booking by changing the check-in and/or check-out dates. Use the AmendmentPriceRequest function to determine the availability and price before booking the amendment.

An amendment will either cancel the existing booking and create a new booking for the new stay (FullPeriod) OR keep the old booking active and book the extra nights in a separate booking (ExtraDays).

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/AmendmentRequest?userName=\[YourUsername\]&password=\[YourPassword\]&bookingID=1454020&roomId=2998&checkInDate=2013-01-01&checkOutDate=2013-01-05&maxPrice=100&language=en&BookingType=FullPeriod](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/AmendmentRequest?userName=[YourUsername]&password=[YourPassword]&bookingID=1454020&roomId=2998&checkInDate=2013-01-01&checkOutDate=2013-01-05&maxPrice=100&language=en&BookingType=FullPeriod)

Function inputs

*: Required input

Parameter name	Description	Valid Input Values
* <i>userName</i>	The name of the user that is performing the amendment operation.	Non-empty strings
* <i>password</i>	The password of the user that is performing the amendment operation.	Non-empty strings
* <i>bookingID</i>	The ID of the booking that is to be amended.	String containing numerical characters
* <i>roomId</i>	The ID of the amendment booking	Integer representing a room ID
* <i>checkInDate</i>	The date for the check-in after amendment.	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number
* <i>checkOutDate</i>	The date for the check-out after amendment.	A date in the format 'YYYY-MM-DD', where YYYY = Year MM = Month number DD = Day number
* <i>maxPrice</i>	The maximum price increase allowed for the amendment.	String containing numerical characters
* <i>language</i>	Determines the language that the response will be displayed in.	A valid ISO code from the "GetLanguages" function.
* <i>bookingType</i>	The type of amendment. Valid values are: "FullPeriod" and "ExtraDays"	Non-empty string

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx?op=AmendmentRequest>

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<amendmentRequestResult xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://xml.sunhotels.net/15/">
  <!--Information about the new amendment booking made.
  XML structure is identical to that of a single booking when using
  GetBookingInformation.
  The admendment booking will always be active.-->
  <AmendmentBooking>
  </AmendmentBooking>
  <!--Information about the original booking that was amended.
  XML structure is identical to that of a single booking when using
  GetBookingInformation.
  The admendment booking will be active when performing an "ExtraDays" amendment
```

```

    and not active (cancelled) when performing a "FullPeriod" amendment.-->
    <OriginalBooking>
  </OriginalBooking>
</amendmentRequestResult>

```

Errors

Listed below are common errors that may appear under the "Error" element.
Should any errors occur that are not listed here, contact Sunhotels.

Error Type	Description	Resolution
AmendmentNotPossibleException	The amendment requested could not be performed.	Check input parameters.
AmendmentAlreadyExistException	An amendment already exists and a new amendment cannot be performed.	Contact Sunhotels to make requested changes.
AmendmentBookingTypeInvalidException	The requested amendment type is invalid.	Use AmendmentPriceRequest to determine the amendment type to use.
AmendmentTransfersNotAllowedException	A transfer is booked on the original booking.	Cancel transfer and rebook after amendment has been completed.
AmendmentBookingTypeChangedException	The requested amendment type is not correct.	Use AmendmentPriceRequest to determine the amendment type to use.
AmendmentPriceOverMaxException	The price of the amendment is higher than the maximum allowed price specified.	Use AmendmentPriceRequest to determine the amendment price.
NonExistentBookingException	The booking selected does not exist.	Use a valid booking id.
NoRoomAvailabilityException	The amendment could not be completed due to lack of availability.	Use AmendmentPriceRequest to determine if amendment is possible.

Cancelling a booking

This function is used to cancel an existing booking.

In case the booking also has a transfer booking attached, the transfer booking will also be cancelled.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/CancelBooking?userName=\[YourUsername\]&password=\[YourPassword\]&bookingID=SH1454020&language=en](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/CancelBooking?userName=[YourUsername]&password=[YourPassword]&bookingID=SH1454020&language=en)

Function inputs

*: Required input

Parameter name	Description	Valid Input Values
* <i>userName</i>	The name of the user that is performing the cancellation operation.	Non-empty strings
* <i>password</i>	The password of the user that is performing the cancellation operation.	Non-empty strings
* <i>bookingID</i>	The ID of the booking that is to be cancelled.	String containing numerical characters, optionally preceded by "SH"
* <i>language</i>	Determines the language that the booking cancellation mail will be displayed in.	A valid ISO code from the "GetLanguages" function.

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx?op=CancelBooking>

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<result>
  <!--The status code of the cancellation. 1 is successful, -1
  is unsuccessful and -2 is booking can't be cancelled/booking don't exists.-->
  <Code></Code>
  <!-- The payment methods (one for each type).
  id: The id of the payment method
  name: The name of the payment method -->
  <CancellationPaymentMethod id="Id" name="Name">
    <!-- The cancellation fees that will be charged when cancelling.
    currency: The currency the amount is expressed in. -->
    <cancellationfee currency="Currency"></cancellationfee>
    <cancellationfee currency="Currency"></cancellationfee>
    <!-- The items that are cancelled (one for each type eg. Hotel, Transfer)
    type: The type of what is cancelled, eg. Hotel, Transfer. -->
    <cancellation type="Type">
      <!-- This element only appears when a cancellation policy applies.
      The active cancellation policy. -->
      <activecancellationpolicy xsi:nil="true"></activecancellationpolicy>
    </cancellation>
    <cancellation type="Type"></cancellation>
  </CancellationPaymentMethod>
  <CancellationPaymentMethod id="Id" name="Name"></CancellationPaymentMethod>
  <!--This element only appears when an error has occurred.-->
  <Error>
    <!--The type of error (exception).-->
    <ErrorType></ErrorType>
    <!--A description of the error.-->
    <Message></Message>
  </Error>
</result>
```


Errors

Listed below are common errors that may appear under the “Error” element.

Should any errors occur that are not listed here, contact Sunhotels.

Error Type	Description	Resolution
BookingCancellationDeadlineExpiredException	The booking could not be cancelled due to the cancellation deadline having expired.	Not available
NonExistentBookingException	The booking to be cancelled could not be found.	Use the “GetBookingInformation” to find the booking to cancel.
BookingAlreadyCancelledException	The booking to be cancelled has already been cancelled.	Not available

Retrieving a list of transfer types

This function is used to retrieve a list of all transfer types and codes (or a single transfer type) in a specified language.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetTransferTypes?userName=\[YourUserName\]&password=\[YourPassword\]&language=en&transferTypeCode=](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetTransferTypes?userName=[YourUserName]&password=[YourPassword]&language=en&transferTypeCode=)

Function inputs

*: Required input

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent performing the search.	Non-empty strings
* <i>password</i>	The password of the agent performing the search.	Non-empty strings
* <i>language</i>	This string will determine the language that the transfer types will be displayed in.	A valid ISO code from the "GetLanguages" function.
<i>transferTypeCode</i>	If specified, only the specified transfer type code will be shown in the results.	Integer representing a transfer type code.

Function outputs

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx?op=GetTransferTypes>

XML Output

```
<?xml version="1.0" encoding="utf-8"?>
<getTransferTypesResult>
  <!--All transfer types that were retrieved.-->
  <transferTypes>
    <transferType>
      <!--The ID of the transfer type.-->
      <id></id>
      <!--The name of the transfer type in the specified language.-->
      <name></name>
    </transferType>
  </transferTypes>
  <!--This element only appears when an error has occurred.-->
  <Error>
    <!--The type of error (the exception type).-->
    <ErrorType></ErrorType>
    <!--A description of the error.-->
    <Message></Message>
  </Error>
</getTransferTypesResult>
```

Retrieving a list of themes

This function is used to retrieve a list of all hotel themes.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetThemes?userName=\[YourUsername\]&password=\[YourPassword\]](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetThemes?userName=[YourUsername]&password=[YourPassword])

Function inputs

*: Required input

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx?op=GetThemes>

XML Output

```
<?xml version="1.0" encoding="utf-8"?>
<getThemesResult>
  <themes>
    <!--id: The ID of the theme
    name: The name of the theme,
    image: The URL to the image associated with the theme-->
    <theme id="1" name="luxury" image="http://www.sunhotels.net/images/luxury.png"
  />
</themes>
<!--This element only appears when an error has occurred.-->
<Error>
  <!--The type of error (the exception type).-->
  <ErrorType></ErrorType>
  <!--A description of the error.-->
  <Message></Message>
</Error>
</getThemesResult>
```

Retrieving a list of destinations

This function is used to retrieve a list of all destinations or destinations with a specific destination code.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/GetDestinations?userName=\[YourUsername\]&password=\[YourPassword\]&language=en&destinationCode=cun&sortBy=Destination&sortOrder=Ascending&exactDestinationMatch=0](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/GetDestinations?userName=[YourUsername]&password=[YourPassword]&language=en&destinationCode=cun&sortBy=Destination&sortOrder=Ascending&exactDestinationMatch=0)

Function inputs

*: Required input

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>language</i>	The language in which the results will be displayed.	A valid ISO code from the "GetLanguages" function.
<i>destinationCode</i>	The destination code that the returned destinations should have. If not specified, all destinations will be displayed.	String representing a destination code, e.g. "CUN".
<i>sortBy</i>	Determines the field that the results should be sorted after. If not specified, the results will be sorted by "Country".	"Country" or "Destination".
<i>sortOrder</i>	Determines the sorting order of the results. If not specified, "Ascending" will be used.	"Ascending" or "Descending".
<i>exactDestinationMatch</i>	If this parameter is set to "true" or "1", only destinations where the searched destination code is the main destination code will be displayed.	"true"/"1" or "false"/"0"

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx?op=GetDestinations>

XML Output

```
<?xml version="1.0" encoding="utf-8"?>
<DestinationList>
  <!--The retrieved destinations-->
  <Destinations>
    <Destination>
      <!--The ID of the destination-->
      <destination_id></destination_id>
      <!--The main destination code (IATA-code) of the destination-->
      <DestinationCode></DestinationCode>
      <!--The second IATA-code of the destination.-->
      <DestinationCode.2></DestinationCode.2>
      <!--The third IATA-code of the destination.-->
      <DestinationCode.3></DestinationCode.3>
      <!--The fourth IATA-code of the destination.-->
      <DestinationCode.4></DestinationCode.4>
      <!--The name of the destination.-->
      <DestinationName></DestinationName>
      <!--The ID of the country where the destination is located.-->
      <CountryId></CountryId>
      <!--The name of the country where the destination is located-->
      <CountryName></CountryName>
      <!--The country code for the country where the destination is located-->
      <CountryCode></CountryCode>
      <!--The time zone for the destination-->
      <TimeZone></TimeZone>
    </Destination>
  </Destinations>
  <!--This element only appears when an error has occurred.-->
  <Error>
    <!--The type of error (the exception type).-->
    <ErrorType></ErrorType>
    <!--A description of the error.-->
    <Message></Message>
  </Error>
</DestinationList>
```

Retrieving a list of resorts

This function is used to retrieve a list of all resorts or resorts with a specific destination code or ID.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetResorts?userName=\[YourUsername\]&password=\[YourPassword\]&language=en&destinationCode=BCN&destinationID=&sortBy=Resort&sortOrder=Ascending&exactDestinationMatch=0](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetResorts?userName=[YourUsername]&password=[YourPassword]&language=en&destinationCode=BCN&destinationID=&sortBy=Resort&sortOrder=Ascending&exactDestinationMatch=0)

Function inputs

*: Required input

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>language</i>	The language in which the results will be displayed.	A valid ISO code from the “GetLanguages” function.
<i>destinationCode</i>	The destination code that the returned resorts should have. If this, along with destinationID, is not specified, all resorts will be displayed.	String representing a destination code, e.g. “CUN”.
<i>destinationID</i>	The destination ID that the returned resorts should have. If this, along with destinationCode, is not specified, all resorts will be displayed.	Integer representing a destination ID
<i>sortBy</i>	Determines the field that the results should be sorted after. If not specified, the results will be sorted by “ResortID”.	“Country”, “Destination”, “ResortID”, “Resort”
<i>sortOrder</i>	Determines the sorting order of the results. If not specified, “Ascending” will be used.	“Ascending” or “Descending”.
<i>exactDestinationMatch</i>	If this parameter is set to “true” or “1”, only resorts where the searched destination code is the main destination code will be displayed. If not specified, all resorts where any of the destination codes are equal to the searched destination code will be shown.	“true”/“1” or “false”/“0”

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx?op=GetResorts>

XML Output

```
<?xml version="1.0" encoding="utf-8"?>
<ResortList>
  <!--All resorts that were retrieved.-->
  <Resorts>
    <Resort>
      <!--The ID of the resort.-->
      <ResortId></ResortId>
      <!--The name of the resort, in the specified language.-->
      <ResortName></ResortName>
      <!--The ID of the destination associated with the resort.-->
      <destination_id></destination_id>
      <!--The name of the associated destination, in the specified
      language.-->
      <DestinationName></DestinationName>
      <!--The name of the country where the resort is located, in the
      specified language.-->
      <CountryName></CountryName>
      <!--The country code of the country where the resort is located-->
      <CountryCode></CountryCode>
    </Resort>
  </Resorts>
  <!--This element only appears when an error has occurred.-->
  <Error>
    <!--The type of error (the exception type).-->
    <ErrorType></ErrorType>
    <!--A description of the error.-->
    <Message></Message>
  </Error>
</ResortList>
```

Transfer searching

This function is used to retrieve a list of available transfer options for a specific room, hotel, booking or resort. You may want to retrieve the transfers from the search result instead of using this function.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/SearchTransfers?userName=\[YourUsername\]&password=\[YourPassword\]&language=en¤cies=GBP,EUR&hotelID=2134&roomID=&bookingID=&resortID=&transferID=&arrivalDate=&arrivalTime=&returnDepartureDate=&returnDepartureTime=](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/SearchTransfers?userName=[YourUsername]&password=[YourPassword]&language=en¤cies=GBP,EUR&hotelID=2134&roomID=&bookingID=&resortID=&transferID=&arrivalDate=&arrivalTime=&returnDepartureDate=&returnDepartureTime=)

Function inputs

*: Required input

**: Required alternative (one, and only one, of these parameters must be specified)

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>language</i>	The language in which the results will be displayed.	A valid ISO code from the "GetLanguages" function.
* <i>currencies</i>	The currencies that the rate prices in the search results will be converted to.	String representing one or more currency codes, e.g. "GBP" or "GBP, USD"
** <i>hotelID</i>	The ID of the hotel.	Integer representing a hotel ID
** <i>roomID</i>	The ID of the room.	Integer representing a room ID
** <i>bookingID</i>	The ID of the booking.	Integer representing a booking ID
** <i>resortID</i>	The ID of the resort.	Integer representing a resort ID
<i>arrivalDate</i>	Date of arrival	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number
<i>arrivalTime</i>	Time of arrival Note: If entered, the " <i>arrivalDate</i> " parameter must also be entered.	A timestamp in the format "hh:mm", where hh = hours and mm = minutes
<i>returnDepartureDate</i>	Date of departure Note: Must occur at a later date than the date specified in the " <i>arrivalDate</i> " parameter (if " <i>arrivalDate</i> " is specified).	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number
<i>returnDepartureTime</i>	Time of departure Note: If entered, the " <i>returnDepartureDate</i> " parameter must also be entered.	A timestamp in the format "hh:mm", where hh = hours and mm = minutes

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx?op=SearchTransfers>

XML Output

```
<?xml version="1.0" encoding="utf-8"?>
<searchTransfersResult>
  <!--All airports that are available for transfers for the given hotel, room or
  booking.-->
  <airports>
    <airport>
      <!--The ID of the airport-->
      <airportId></airportId>
      <!--The ID of the resort associated with the airport.-->
      <resortId></resortId>
      <!--The name of the airport-->
      <airportName></airportName>
      <!--The IATA code of the airport.-->
      <iataCode></iataCode>
      <!--The time zone of the destination where the airport is located.-->
      <timeZone></timeZone>
      <!--All available transfers for the airport.-->
      <transfers>
        <transfer>
          <!--The rates for the transfer-->
          <rates>
            <rate>
              <!--Whether the rate applies to an arrival transfer
              or a return transfer-->
              <direction>Arrival or Return</direction>
              <!--The hour of day when the rate becomes active-->
              <startHour></startHour>
              <!--The hour of day when the rate is no longer active-->
              <endHour></endHour>
              <!--The rate prices (in the specified currencies) -->
              <prices>
                <!--currency: The currency code that the rate price applies to-->
                <price currency="">
                  <!--The amount is stored as content of the
                  "price" element-->
                  </price>
                </prices>
              </rate>
            </rates>
            <!--The ID of the transfer.-->
            <transferId></transferId>
            <!--The ID of the transfer type.-->
            <transferTypeId></transferTypeId>
            <!--The minimum amount of passengers allowed for the transfer.-->
            <minimumPassengers></minimumPassengers>
            <!--The maximum amount of passengers allowed for the transfer.-->
            <maximumPassengers></maximumPassengers>
            <!--The estimated time for the transfer, in hours.-->
            <estimatedTimeInHours></estimatedTimeInHours>
            <!--The estimated time for the transfer, in minutes.-->
            <estimatedTimeInMinutes></estimatedTimeInMinutes>
            <!--The name of the transfer type.-->
            <transferTypeName></transferTypeName>
          </transfer>
        </transfers>
      </airport>
    </airports>
    <!--This element only appears when an error has occurred.-->
    <Error>
      <!--The type of error (the exception type).-->
      <ErrorType></ErrorType>
```

```
<!--A description of the error.-->
<Message></Message>
</Error>
</searchTransfersResult>
```

Transfer searching V2

This function is used to retrieve a list of available transfer for a specific room, hotel, booking, Hotel GIATA code or resort. (In case that the search do not return results using the Hotel GIATA code is recommendable search in those cases using the resort related to that hotel).

Example URL with input:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/SearchTransfersV2?userName=%5bYourUsername%5d&password=%5bYourPassword%5d&language=en¤cies=GBP,EUR&giataCode=XXXX&hotelID=&roomID=&bookingID=&resortID=&transferID=&arrivalDate=&arrivalTime=&returnDepartureDate=&returnDepartureTime=>

Function inputs

*: Required input

**: Required alternative (one, and only one, of these parameters must be specified)

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>language</i>	The language in which the results will be displayed.	A valid ISO code from the "GetLanguages" function.
* <i>currencies</i>	The currencies that the rate prices in the search results will be converted to.	String representing one or more currency codes, e.g. "GBP" or "GBP, USD"
** <i>hotelID</i>	The ID of the hotel.	Integer representing a hotel ID
** <i>roomID</i>	The ID of the room.	Integer representing a room ID
** <i>bookingID</i>	The ID of the booking.	Integer representing a booking ID
** <i>resortID</i>	The ID of the resort.	Integer representing a resort ID
** <i>transferID</i>	The ID of the transfer.	Integer representing a transfer ID
<i>arrivalDate</i>	Date of arrival	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number
<i>arrivalTime</i>	Time of arrival Note: If entered, the " <i>arrivalDate</i> " parameter must also be entered.	A timestamp in the format "hh:mm", where hh = hours and mm = minutes

<i>returnDepartureDate</i>	Date of departure Note: Must occur at a later date than the date specified in the “ <i>arrivalDate</i> ” parameter (if “ <i>arrivalDate</i> ” is specified).	A date in the format ‘YYYY-MM-DD’ YYYY = Year MM = Month number DD = Day number
<i>returnDepartureTime</i>	Time of departure Note: If entered, the “ <i>returnDepartureDate</i> ” parameter must also be entered.	A timestamp in the format “hh:mm”, where hh = hours and mm = minutes
<i>**giataCode</i>	The GIATA Code of the property	Integer representing the GIATA code of a property

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx?op=SearchTransfersV2>

XML Output

```
<?xml version="1.0" encoding="utf-8"?>
<searchTransfersResult>
  <!--All airports that are available for transfers for the given hotel, room or booking.-->
  <airports>
    <airport>
      <!--The ID of the airport-->
      <airportId></airportId>
      <!--The ID of the resort associated with the airport.-->
      <resortId></resortId>
      <!--The name of the airport-->
      <airportName></airportName>
      <!--The IATA code of the airport.-->
      <iataCode></iataCode>
      <!--The time zone of the destination where the airport is located.-->
      <timeZone></timeZone>
      <!--All available transfers for the airport.-->
      <transfers>
        <transfer>
          <!--The rates for the transfer-->
          <rates>
            <rate>
              <!--Whether the rate applies to an arrival transfer or a return transfer-->
              <direction>Arrival or Return</direction>
              <!--The hour of day when the rate becomes active-->
              <startHour></startHour>
              <!--The hour of day when the rate is no longer active-->
              <endHour></endHour>
              <!--The rate prices (in the specified currencies) -->
              <prices>
                <!--currency: The currency code that the rate price applies to-->
                <price currency="">
                  <!--The amount is stored as content of the "price" element-->
                  </price>
                </prices>
              </rate>
            </rates>
            <!--The ID of the transfer.-->
            <transferId></transferId>
            <!--The ID of the transfer type.-->
```

```
<transferTypeId></transferTypeId>
<!--The minimum amount of passengers allowed for the transfer.-->
<minimumPassengers></minimumPassengers>
<!--The maximum amount of passengers allowed for the transfer.-->
<maximumPassengers></maximumPassengers>
<!--The estimated time for the transfer, in hours.-->
<estimatedTimeInHours></estimatedTimeInHours>
<!--The estimated time for the transfer, in minutes.-->
<estimatedTimeInMinutes></estimatedTimeInMinutes>
<!--The name of the transfer type.-->
<transferTypeName></transferTypeName>
</transfer>
</transfers>
</airport>
</airports>
<!--This element only appears when an error has occurred.-->
<Error>
  <!--The type of error (the exception type).-->
  <ErrorType></ErrorType>
  <!--A description of the error.-->
  <Message></Message>
</Error>
</searchTransfersResult>
```

Retrieving a list of static hotels and rooms

This function is used to retrieve detailed information about a specific hotel or all hotels on a specific destination or resort.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/GetStaticHotelsAndRooms?userName=\[YourUsername\]&password=\[YourPassword\]&language=en&destination=&hotelIDs=6928&resortIDs=&accommodationTypes=&sortBy=&sortOrder=&exactDestinationMatch=](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/GetStaticHotelsAndRooms?userName=[YourUsername]&password=[YourPassword]&language=en&destination=&hotelIDs=6928&resortIDs=&accommodationTypes=&sortBy=&sortOrder=&exactDestinationMatch=)

Function inputs

*: Required input

**: Required alternative (one, and only one, of these parameters must be specified)

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>language</i>	The language in which the results will be displayed.	A valid ISO code from the "GetLanguages" function.
** <i>destination</i>	IATA-code for the airport that the hotels should be associated with	String value representing an IATA-code
** <i>hotelIDs</i>	The specific ID(s) of the hotel(s)	A string containing hotel IDs, separated by commas, e.g. "12,34,56"
** <i>resortIDs</i>	The specific ID(s) of the resort(s) where the hotels should be located.	A string containing resort IDs, separated by commas. Example: "12,34,56"
<i>accommodationTypes</i>	The types of accommodation that is to be searched. If not specified, all types will be searched.	"Hotel", "Apartment", "Villa"
<i>sortBy</i>	The field that the results should be sorted after. If not specified, "Hotel" is used.	"Hotel", "Resort", "Classification"
<i>sortOrder</i>	Determines the sorting order of the results. If not specified, "Ascending" will be used.	"Ascending" or "Descending"
<i>exactDestinationMatch</i>	If set to "true" or "1" and "destination" is set, the search results will only contain rooms where the destination code is the primary destination code.	"true"/"1" or "false"/"0"

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx?op=GetStaticHotelsAndRooms>

XML Output

```
<?xml version="1.0" encoding="utf-8"?>
<getStaticHotelsAndRoomsResult xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://xml.sunhotels.net/15/">
  <hotels>
    <hotel>
      <!--The unique ID of the hotel-->
      <hotel.id></hotel.id>
      <!--The ID of the destination where the hotel is located-->
      <destination.id></destination.id>
```

```

<!--The ID of the resort where the hotel is located-->
<resort_id></resort_id>
<!--Integer value indicating whether it is possible to add transfer
  1 = adding transfer possible, 0 = adding transfer not possible-->
<transfer></transfer>
<!--All room types that are available for the accommodation.-->
<roomtypes>
  <roomtype xsi:type="StaticRoomTypeWithRooms">
    <!--The ID of the room type.-->
    <roomtype.ID></roomtype.ID>
    <!--All rooms for the room type that belong
    to the accommodation and match the search criteria-->
    <rooms>
      <room xsi:type="StaticRoom">
        <!--The ID of the room-->
        <id></id>
        <!--The number of bed spaces that are available for the room-->
        <beds></beds>
        <!--The number of extra bed spaces that are available for the room-->
        <extrabeds></extrabeds>
        <!--Always empty for this function-->
        <meals></meals>
        <!--Always empty for this function-->
        <cancellation_policies></cancellation_policies>
        <!--Always empty for this function-->
        <notes></notes>
        <!--Determines whether the room is a "Super Deal" room-->
        <isSuperDeal></isSuperDeal>
        <!--Determines whether the room is a "Best Buy" room-->
        <isBestBuy></isBestBuy>
        <!--The available payment methods for the room-->
        <paymentMethods>
          <!--id: The ID of the payment method. 1: Invoice/Pay
now, 2: Pay at hotel
          name: The translated name of the payment method.
          For more information about payment methods, visit
          http://xml.sunhotels.net/15/PaymentMethods/ -->
          <paymentMethod xsi:type="StaticPaymentMethod" id="" name="">
            <!--Depending on the payment method,
            different properties may appear here.-->
            <property key="" value=""/>
            <property key="" value=""/>
          </paymentMethod>
        </paymentMethods>
        <!--The features of the room-->
        <features>
          <!--id: The id of the feature, name: The name of the feature-->
          <feature id="" name="" />
        </features>
        <!--The themes of the room-->
        <themes>
          <!--
id: The id of the theme, name: The name of the theme, image: The image associated
with the theme-->
          <theme id="" name="" image="" />
        </themes>
      </room>
    </rooms>
    <!--The name of the room type (translated to the language

```

```

        specified for the current search)-->
        <room.type></room.type>
        <room.type></room.type>
        <!-- Whether rooms are shared for this room type, if true minimum age is
18 years. -->
        <sharedRoom></sharedRoom>
        <!-- Whether facilities are shared for this room type, if true minimum
age is 18 years. -->
        <sharedFacilities></sharedFacilities>
    </roomtype>
</roomtypes>
<!--Contains review information for this accommodation-->
<review>
    <!--Average rating-->
    <rating></rating>
    <!--Average price value-->
    <pricevalue></pricevalue>
    <!--Number of reviews-->
    <count></count>
    <!--Number of recommendations for the accommodation-->
    <recommendCount></recommendCount>
</review>
<!--Always empty for this function-->
<notes />
<!--Always empty for this function-->
<isPrioritized xsi:nil="true" />
<!--Type of accommodation, e.g. "Hotel", "Apartment" or "Villa"-->
<type></type>
<!--The name of the accommodation-->
<name></name>
<!--First address line of the accommodation-->
<hotel.addr.1></hotel.addr.1>
<!--Second address line of the accommodation-->
<hotel.addr.2></hotel.addr.2>
<!--Zip code of the accommodation-->
<hotel.addr.zip></hotel.addr.zip>
<!--The city where the accommodation is located-->
<hotel.addr.city></hotel.addr.city>
<!--The state where the accommodation is located-->
<hotel.addr.state></hotel.addr.state>
<!--The country where the accommodation is located-->
<hotel.addr.country></hotel.addr.country>
<!--The country code of the country where the accommodation is located-->
<hotel.addr.countrycode></hotel.addr.countrycode>
<!--The complete address of the accommodation-->
<hotel.address></hotel.address>
<!--The URL to a map view of the accommodation-->
<hotel.mapurl></hotel.mapurl>
<!--The accommodation headline-->
<headline></headline>
<!--The description of the accommodation-->
<description></description>
<!--The name of the resort where the accommodation is located-->
<resort></resort>
<!--The name of the destination where the accommodation is located.-->
<destination></destination>
<!--The themes of the accommodation-->
<themes>
    <!--

```

```

id: The id of the theme, name: The name of the theme, image: The image associated
with the theme-->
  <theme id="" name="" image="" />
</themes>
<!--All images for the accommodation-->
<images>
  <!--The id of the image-->
  <image id="">
    <!--The full-size version of the image
    url: The url to the image
    (needs to be prepended with "https://hotelimages.sunhotels.net/HotelInfo
/hotelImage.aspx?id=10473272&full=1")
    height: The height of the image (in pixels)
    width: The width of the image (in pixels)-->
    <fullSizeImage url="" height="" width="" />
    <!--The small version of the image
    url: The url to the image
    (needs to be prepended with "https://hotelimages.sunhotels.net/HotelInfo
/hotelImage.aspx?id=10473272")
    height: The height of the image (in pixels)
    width: The width of the image (in pixels)-->
    <smallImage url="" height="" width="" />
  </image>
</images>
<!--The features of the accomodation-->
<features>
  <!--id: The id of the feature, name: The name of the feature-->
  <feature id="" name="" />
</features>
<!--The accommodation classification,
e.g. a digit between 1 and 5 plus an optional character, e.g. "+"-->
<classification></classification>
<!--The coordinates for the accommodation
(if the agent is allowed to view coordinates).-->
<coordinates>
  <!--The latitude coordinate for the accommodation.-->
  <latitude></latitude>
  <!--The longitude coordinate for the accommodation.-->
  <longitude></longitude>
</coordinates>
<!--A list of distance types linked to the accommodation,
i.e. distances to various beaches, airports, etc.-->
<distanceTypes>
  <distanceType>
    <!--The ID of the distance type-->
    <hotelDistanceTypeId></hotelDistanceTypeId>
    <!--The description of the type of place the child distances refer to-->
    <description></description>
    <distances>
      <distance>
        <!--The distance (in meters)-->
        <distanceInMeters></distanceInMeters>
        <!--The name of a place-->
        <placeName></placeName>
        <!--Coordinates for the described place-->
        <coordinates>
          <!--The latitude coordinate-->
          <latitude></latitude>
          <!--The longitude coordinate-->

```



```
        <longitude></longitude>
      </coordinates>
    </distance>
  </distances>
</distanceType>
</distanceTypes>
<!--The time zone where the accommodation is located-->
<timeZone></timeZone>
<!--Whether accommodation is "Best Buy" or not.-->
<isBestBuy></isBestBuy>
<codes>
  <!-- Codes.-->
  <code type="" value=""></code>
</codes>
</hotel>
</hotels>
<!--This element only appears when an error has occurred.-->
<Error>
  <!--The type of error (exception).-->
  <ErrorType></ErrorType>
  <!--A description of the error.-->
  <Message></Message>
</Error>
</getStaticHotelsAndRoomsResult>
```

Retrieving booking information

This function is used to get information about an existing booking or all bookings created within a specific date range or all bookings with an arrival date within a specific date range.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/GetBookingInformation?userName=\[Your Username\]&password=\[YourPassword\]&language=en&bookingID=1454020&reference=&createdDateFrom=&createdDateTo=&arrivalDateFrom=&arrivalDateTo=](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/GetBookingInformation?userName=[Your Username]&password=[YourPassword]&language=en&bookingID=1454020&reference=&createdDateFrom=&createdDateTo=&arrivalDateFrom=&arrivalDateTo=)

Function inputs

*: Required input

**: Required alternative (one of these parameters must be specified)

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>language</i>	The language in which the results will be displayed.	A valid ISO code from the "GetLanguages" function.
** <i>bookingID</i>	The ID of an existing booking.	Integers > 0
** <i>reference</i>	A reference associated with an existing booking	String representing a booking reference.
** <i>createdDateFrom</i>	Specifies the start date of the time period when the searched bookings are to have been created. Note: If specified, <i>createdDateTo</i> must also be specified.	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number
** <i>createdDateTo</i>	Specifies the start date of the time period when the searched bookings are to have been created. Note: If specified, <i>createdDateFrom</i> must also be specified.	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number
** <i>arrivalDateFrom</i>	Specifies the start of the date range when the arrival date has occurred. Note: If specified, <i>arrivalDateTo</i> must also be specified.	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number
** <i>arrivalDateTo</i>	Specifies the end of the date range when the arrival date has occurred. Note: If specified, <i>arrivalDateFrom</i> must also be specified.	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx?op=GetBookingInformation>

XML Output

```
<?xml version="1.0" encoding="utf-8" ?>
<getBookingInformationResult>
  <!--All retrieved bookings-->
  <bookings>
    <booking>
      </booking>
    </bookings>
  <!--This element only appears when an error has occurred.-->
  <Error>
    <!--The type of error (the exception type).-->
    <ErrorType></ErrorType>
    <!--A description of the error.-->
    <Message></Message>
  </Error>
</getBookingInformationResult>
```

Retrieving booking information V2

This function is used to get information about an existing booking or all bookings created within a specific date range or all bookings with an arrival date within a specific date range. Allows to return guests names using the showGuests parameter.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetBookingInformationV2?userName=\[YourUsername\]&password=\[YourPassword\]&language=en&bookingID=1454020&reference=&createdDateFrom=&createdDateTo=&arrivalDateFrom=&arrivalDateTo=&showGuests=](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetBookingInformationV2?userName=[YourUsername]&password=[YourPassword]&language=en&bookingID=1454020&reference=&createdDateFrom=&createdDateTo=&arrivalDateFrom=&arrivalDateTo=&showGuests=)

Function inputs

*: Required input

**: Required alternative (one of these parameters must be specified)

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>language</i>	The language in which the results will be displayed.	A valid ISO code from the "GetLanguages" function.
** <i>bookingID</i>	The ID of an existing booking.	Integers > 0
** <i>reference</i>	A reference associated with an existing booking	String representing a booking reference.
** <i>createdDateFrom</i>	Specifies the start date of the time period when the searched bookings are to have been created. Note: If specified, <i>createdDateTo</i> must also be specified.	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number
** <i>createdDateTo</i>	Specifies the start date of the time period when the searched bookings are to have been created. Note: If specified, <i>createdDateFrom</i> must also be specified.	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number

**arrivalDateFrom	Specifies the start of the date range when the arrival date has occurred. Note: If specified, <i>arrivalDateTo</i> must also be specified.	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number
**arrivalDateTo	Specifies the end of the date range when the arrival date has occurred. Note: If specified, <i>arrivalDateFrom</i> must also be specified.	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number
showGuests	Determines whether the guests names should be displayed.	"true"/"1" to display guests names.

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx?op=GetBookingInformationV2>

XML Output

```
<?xml version="1.0" encoding="utf-8" ?>
<getBookingInformationResult>
  <!--All retrieved bookings-->
  <bookings>
    <booking>
    </booking>
  </bookings>
  <!--This element only appears when an error has occurred.-->
  <Error>
    <!--The type of error (the exception type).-->
    <ErrorType></ErrorType>
    <!--A description of the error.-->
    <Message></Message>
  </Error>
</getBookingInformationResult>
```

Retrieving booking information V3

The GetBookingInformationV3 method includes all the parameters from the previous version but the response returns **Taxes** and **Fees**.

XML Output

```
<?xml version="1.0" encoding="utf-8" ?>
<getBookingInformationResult>
  <!--All retrieved bookings-->
  <bookings>
    <booking>
    </booking>
  </bookings>
  <!--This element only appears when an error has occurred.-->
  <Error>
    <!--The type of error (the exception type).-->
    <ErrorType></ErrorType>
    <!--A description of the error.-->
    <Message></Message>
  </Error>
</getBookingInformationResult>
```

Retrieving transfer booking information

This function is used to get information about a transfer add-on booking.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetTransferBookingInformation?userName=\[YourUsername\]&password=\[YourPassword\]&language=en&bookingID=2435578&createdDateFrom=&createdDateTo=&arrivalDateFrom=&arrivalDateTo=](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetTransferBookingInformation?userName=[YourUsername]&password=[YourPassword]&language=en&bookingID=2435578&createdDateFrom=&createdDateTo=&arrivalDateFrom=&arrivalDateTo=)

Function inputs

*: Required input

**: Required alternative (one of these parameters must be specified)

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>language</i>	The language in which the results will be displayed.	A valid ISO code from the "GetLanguages" function.
* <i>bookingID</i>	The ID of an existing booking.	Integers > 0
** <i>createdDateFrom</i>	Specifies the start date of the time period when the searched bookings are to have been created. Note: If specified, <i>createdDateTo</i> must also be specified.	A date in the format 'YYYY-MM-DD" YYYY = Year MM = Month number DD = Day number
** <i>createdDateTo</i>	Specifies the start date of the time period when the searched bookings are to have been created. Note: If specified, <i>createdDateFrom</i> must also be specified.	A date in the format 'YYYY-MM-DD" YYYY = Year MM = Month number DD = Day number
** <i>arrivalDateFrom</i>	Specifies the start of the date range when the arrival date has occurred. Note: If specified, <i>arrivalDateTo</i> must also be specified.	A date in the format 'YYYY-MM-DD" YYYY = Year MM = Month number DD = Day number
** <i>arrivalDateTo</i>	Specifies the end of the date range when the arrival date has occurred. Note: If specified, <i>arrivalDateFrom</i> must also be specified.	A date in the format 'YYYY-MM-DD" YYYY = Year MM = Month number DD = Day number

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx?op=GetTransferBookingInformation>

XML Output

```
<?xml version="1.0" encoding="utf-8"?>
<getTransferBookingInformationResult>
  <!--All transfer add-on bookings that were retrieved for the provided booking.-->
  <transferBookings>
    <transferBooking>
      <!--The ID of the booking.-->
      <transferbookingnumber></transferbookingnumber>
      <!--The ID of the hotel where the transfer is booked-->
      <hotel.id></hotel.id>
      <!--The name of the hotel where the transfer is booked.-->
      <hotel.name></hotel.name>
      <!--The name of the contact person for the transfer booking.-->
      <contactPerson></contactPerson>
      <!--The contact cellphone number for the transfer booking.-->
      <contactCellphone></contactCellphone>
      <!--The airline used.-->
      <airline></airline>
      <!--The flight number used.-->
      <flightNumber></flightNumber>
      <!--The airport used for departure.-->
      <departureAirport></departureAirport>
      <!--The airport used for arrival.-->
      <arrivalAirport></arrivalAirport>
      <!--The time of departure (in the format hh:mm, where hh = hours,
      ranging from 00 to 23, mm = minutes, ranging from 00 to 59).-->
      <departureTime></departureTime>
      <!--The date of arrival (in the format YYYY-MM-DD,
      where YYYY = year, MM = month number and DD = day number).-->
      <arrivalDate></arrivalDate>
      <!--The time of arrival (in the format hh:mm, where hh = hours,
      ranging from 00 to 23, mm = minutes, ranging from 00 to 59).-->
      <arrivalTime></arrivalTime>
      <!--The number of passengers.-->
      <passengers></passengers>
      <!--The ID of the associated transfer.-->
      <transferID></transferID>
      <!--The ID of the transfer type.-->
      <transferTypeID></transferTypeID>
      <!--The name of the transfer type.-->
      <transferType></transferType>
      <!--Text that is shown on the arrival voucher.-->
      <arrivalVoucherText></arrivalVoucherText>
      <!--Text that is shown on the return voucher.-->
      <returnVoucherText></returnVoucherText>
      <!--The URL for the voucher.-->
      <voucherURL></voucherURL>
      <!--The provider from the transfer voucher.-->
      <provider></provider>
      <!--The phone numbers from the transfer voucher.-->
      <providerPhone></providerPhone>
      <!--The emergency phone numbers from the transfer voucher.-->
      <providerEmergencyPhone></providerEmergencyPhone>
      <!--Integer indicating whether a return transfer is included.
      1 = return transfer included, 0 = return transfer not included.-->
      <returntransfer></returntransfer>
      <!--The airline used for the return transfer.-->
      <returnairline></returnairline>
```

```
<!--The flight number used for the return transfer.-->
<returnflightNumber></returnflightNumber>
<!--The date of departure for the return transfer.-->
<returndepartureDate></returndepartureDate>
<!--The time of departure for the return transfer.-->
<returndepartureTime></returndepartureTime>
<!--The price for the transfer in the used currency,.-->
<price></price>
<!--The used currency.-->
<currency></currency>
<!--The price for the transfer in Euros.-->
<price.EUR></price.EUR>
<!--The status of the booking. "Status: Active" means the booking is confirmed in our
system although is flagged as failed (timeout)-->
<bookingstatus>NotActive or Active</bookingstatus>
<!--The invoice reference used when booking-->
<invoiceref></invoiceref>
<!--The reference used when booking-->
<yourref></yourref>
</transferBooking>
</transferBookings>
<!--This element only appears when an error has occurred.-->
<Error>
  <!--The type of error (the exception type).-->
  <ErrorType></ErrorType>
  <!--A description of the error.-->
  <Message></Message>
</Error>
</getTransferBookingInformationResult>
```

Booking a transfer

This function is used to book a transfer, either to an existing booking or stand-alone.

Adding a transfer to an existing hotel booking is allowed up to 48 hours before hotel arrive date (12 noon - hotel local time).

Transfer bookings are non-refundable upon cancellation within 48 hours before the hotel check-in date (12 noon – hotel/resort local time).

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/AddTransfer?userName=\[YourUsername\]&password=\[YourPassword\]&language=en¤cy=EUR&email=noreply@sunhotels.net&bookingID=8000321&hotelName=&contactPerson=testcontactperson&contactCellphone=testcellphone&airline=testairline&flightNumber=testflightnumber&departureAirport=testdepartureairport&departureTime=12:23&arrivalTime=23:48&arrivalDate=2007-05-02&passengers=3&transferID=518&returnTransfer=1&returnAirline=testreturnairline&returnFlightNumber=testreturnflightnumber&returnDepartureDate=2007-07-07&returnDepartureTime=13:45&invoiceRef=&yourRef=](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/AddTransfer?userName=[YourUsername]&password=[YourPassword]&language=en¤cy=EUR&email=noreply@sunhotels.net&bookingID=8000321&hotelName=&contactPerson=testcontactperson&contactCellphone=testcellphone&airline=testairline&flightNumber=testflightnumber&departureAirport=testdepartureairport&departureTime=12:23&arrivalTime=23:48&arrivalDate=2007-05-02&passengers=3&transferID=518&returnTransfer=1&returnAirline=testreturnairline&returnFlightNumber=testreturnflightnumber&returnDepartureDate=2007-07-07&returnDepartureTime=13:45&invoiceRef=&yourRef=)

Function inputs

*: Required input

**: Required alternative (one of these parameters must be specified)

NOTE:

"When using SOAP the arrival date and arrival time is sent using one parameter "arrivalTime" containing both date and time."

"When using SOAP the return date and return time is sent using one parameter "returnDepartureTime" containing both date and time."

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>language</i>	The language in which the results will be displayed.	A valid ISO code from the "GetLanguages" function.
* <i>currency</i>	The currency that all prices will be converted to in the search results.	String representing a currency code, e.g. "GBP".
* <i>email</i>	The e-mail address of the agent. This is where the booking confirmation e-mail is sent.	String representing a valid e-mail address.
** <i>bookingID</i>	The ID of an existing booking.	Integers > 0
** <i>hotelName</i>	The name of an existing hotel	String representing the name of a hotel
* <i>contactPerson</i>	The surname and last name of the contact person	String representing a surname and last name
* <i>contactCellphone</i>	The contact cellphone number	String representing a cell phone number
* <i>airline</i>	The name of the airline used for the arrival transfer.	String representing the name of an airline
* <i>flightNumber</i>	The flight number used for the arrival transfer	String representing a flight number
* <i>departureAirport</i>	The name or IATA-code of the airport used for departing to the booked hotel	String representing the name or IATA-code of an airport
* <i>departureTime</i>	Time of departure	A timestamp in the format "hh:mm", where hh = hours and mm = minutes

<i>*arrivalTime</i>	Time of arrival	A timestamp in the format "hh:mm", where hh = hours and mm = minutes
<i>*arrivalDate</i>	Date of arrival	A date in the format "YYYY-MM-DD" YYYY = Year MM = Month number DD = Day number
<i>*passengers</i>	The number of passengers to transfer	Integers > 0
<i>*transferID</i>	The ID of the associated transfer	Integer representing the ID of a transfer
<i>*returnTransfer</i>	Indicates whether a return transfer should be booked	0 = only arrival transfer or 1 = both arrival and return transfer or 2 = only return transfer
<i>*returnAirline</i>	The name of the airline used for the return transfer Note: This is only required if "returnTransfer" is set to 1	String representing the name of an airline
<i>*returnFlightNumber</i>	The flight number used for the return transfer Note: This is only required if "returnTransfer" is set to 1	String representing a flight number
<i>*returnDepartureDate</i>	Date of departure Note: This is only required if "returnTransfer" is set to 1	A date in the format "YYYY-MM-DD" YYYY = Year MM = Month number DD = Day number
<i>*returnDepartureTime</i>	Time of departure Note: This is only required if "returnTransfer" is set to 1	A timestamp in the format "hh:mm", where hh = hours and mm = minutes
<i>invoiceRef</i>	An invoice reference.	A string representing your invoice reference number.
<i>yourRef</i>	A booking number or reference.	A string representing your reference.

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx?op=AddTransfer>

XML Output

```
<?xml version="1.0" encoding="utf-8"?>
<TransferBookingList>
  <!--This list will only contain information about the transfer that has been
booked.-->
  <transferbookings>
    <transferbooking>
      <!--The ID of the booking where the transfer was added.-->
      <transferbookingnumber></transferbookingnumber>
      <!--The ID of the hotel where the transfer is booked-->
      <hotel.id></hotel.id>
      <!--The name of the hotel where the transfer is booked.-->
      <hotel.name></hotel.name>
      <!--The name of the contact person for the transfer booking.-->
      <contactPerson></contactPerson>
```

```

<!--The contact cellphone number for the transfer booking.-->
<contactCellphone></contactCellphone>
<!--The airline used.-->
<airline></airline>
<!--The flight number used.-->
<flightNumber></flightNumber>
<!--The airport used for departure.-->
<departureAirport></departureAirport>
<!--The airport used for arrival.-->
<arrivalAirport></arrivalAirport>
<!--The time of departure (in the format hh:mm, where hh = hours,
ranging from 00 to 23, mm = minutes, ranging from 00 to 59).-->
<departureTime></departureTime>
<!--The date of arrival (in the format YYYY-MM-DD,
where YYYY = year, MM = month number and DD = day number).-->
<arrivalDate></arrivalDate>
<!--The time of arrival (in the format hh:mm, where hh = hours,
ranging from 00 to 23, mm = minutes, ranging from 00 to 59).-->
<arrivalTime></arrivalTime>
<!--The number of passengers.-->
<passengers></passengers>
<!--The ID of the associated transfer.-->
<transferID></transferID>
<!--The ID of the transfer type.-->
<transferTypeID></transferTypeID>
<!--The name of the transfer type.-->
<transferType></transferType>
<!--Text that is shown on the arrival voucher.-->
<arrivalVoucherText></arrivalVoucherText>
<!--Text that is shown on the return voucher.-->
<returnVoucherText></returnVoucherText>
<!--The URL for the voucher.-->
<voucherURL></voucherURL>
<!--The provider from the transfer voucher.-->
<provider></provider>
<!--The phone numbers from the transfer voucher.-->
<providerPhone></providerPhone>
<!--The emergency phone numbers from the transfer voucher.-->
<providerEmergencyPhone></providerEmergencyPhone>
<!--Integer indicating whether a return transfer is included.
1 = return transfer included, 0 = return transfer not included.-->
<returntransfer></returntransfer>
<!--The airline used for the return transfer.-->
<returnairline></returnairline>
<!--The flight number used for the return transfer.-->
<returnflightNumber></returnflightNumber>
<!--The date of departure for the return transfer.-->
<returndepartureDate></returndepartureDate>
<!--The time of departure for the return transfer.-->
<returndepartureTime></returndepartureTime>
<!--The price for the transfer in the used currency.-->
<price></price>
<!--The used currency.-->
<currency></currency>
<!--The price for the transfer in Euros.-->
<price.EUR></price.EUR>
<!--The status of the booking. -->
<bookingstatus>NotActive or Active</bookingstatus>
<!--The invoice reference used when booking.-->
<invoiceref></invoiceref>
<!-- The reference used when booking.-->
<yourref></yourref>

```

```

    </transferbooking>
  </transferbookings>
  <!--This element only appears when an error has occurred.-->
  <Error>
    <!--The type of error (the exception type).-->
    <ErrorType></ErrorType>
    <!--A description of the error.-->
    <Message></Message>
  </Error>
</TransferBookingList>

```

Errors

Listed below are common errors that may appear under the “Error” element.
Should any errors occur that are not listed here, contact Sunhotels.

Error Type	Description	Resolution
InvalidParametersForTransferException	You are trying to book an arrival-only transfer but have provided return-specific parameters to the function.	Remove the values for the return-specific parameters.
NoTransfersAvailableForCCAgentsException	You are trying to book a transfer while the “force_creditcard” setting for your account is set to “1”, which is not possible.	N/A
NonExistentTransferException	You are trying to book a transfer that does not exist.	Use the “SearchTransfers” function to find available transfers to book.
BookingNotActiveException	You are trying to add a transfer to a hotel booking that is not active for some reason (e.g. have been cancelled, etc.)	Add the transfer to a different booking or as a stand-alone transfer booking.
PassengersLessThanTransferMinimumException	The provided number of passengers is less than the minimum allowed number of passengers for the transfer.	Increase the number of passengers to meet the minimum criterion, or try to book a different transfer.
PassengersMoreThanTransferMaximumException	The provided number of passengers is more than the maximum allowed number of passengers for the transfer.	Decrease the number of passengers to meet the minimum criterion, or try to book a different transfer.
TransferAlreadyBookedException	You are trying to add a transfer to a booking that already has a transfer added to it.	Add the transfer to a different booking or cancel the existing transfer booking.

Booking a transfer V2

This function is used to book a transfer, either to an existing booking or stand-alone.

Adding a transfer to an existing hotel booking is allowed up to 48 hours before hotel arrive date (12 noon - hotel local time).

Transfer bookings are non-refundable upon cancellation within 48 hours before the hotel check-in date (12 noon – hotel/resort local time).

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/AddTransferV2?userName=\[YourUsername\]&password=\[YourPassword\]&language=en¤cy=EUR&email=noreply@sunhotels.net&bookingID=8000321&hotelName=&contactPerson=testcontactperson&contactCellphone=testcellphone&airline=testairline&flightNumber=testflightnumber&departureAirport=testdepartureairport&departureTime=12:23&arrivalTime=23:48&arrivalDate=2007-05-02&passengers=3&transferID=518&returnTransfer=1&returnAirline=testreturnairline&returnFlightNumber=testreturnflightnumber&returnDepartureDate=2007-07-07&returnDepartureTime=13:45&invoiceRef=&yourRef=&departureIataCode=&arrivalIataCode=](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx/AddTransferV2?userName=[YourUsername]&password=[YourPassword]&language=en¤cy=EUR&email=noreply@sunhotels.net&bookingID=8000321&hotelName=&contactPerson=testcontactperson&contactCellphone=testcellphone&airline=testairline&flightNumber=testflightnumber&departureAirport=testdepartureairport&departureTime=12:23&arrivalTime=23:48&arrivalDate=2007-05-02&passengers=3&transferID=518&returnTransfer=1&returnAirline=testreturnairline&returnFlightNumber=testreturnflightnumber&returnDepartureDate=2007-07-07&returnDepartureTime=13:45&invoiceRef=&yourRef=&departureIataCode=&arrivalIataCode=)

Function inputs

*: Required input

**: Required alternative (one of these parameters must be specified)

NOTE:

"When using SOAP the arrival date and arrival time is sent using one parameter "arrivalTime" containing both date and time."

"When using SOAP the return date and return time is sent using one parameter "returnDepartureTime" containing both date and time."

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>language</i>	The language in which the results will be displayed.	A valid ISO code from the "GetLanguages" function.
* <i>currency</i>	The currency that all prices will be converted to in the search results.	String representing a currency code, e.g. "GBP".
* <i>email</i>	The e-mail address of the agent. This is where the booking confirmation e-mail is sent.	String representing a valid e-mail address.
** <i>bookingID</i>	The ID of an existing booking.	Integers > 0
** <i>hotelID</i>	The ID of an existing hotel.	Integer representing a hotel ID
** <i>hotelGiataCode</i>	The GIATA Code of the property	Integer representing the GIATA code of a property
** <i>hotelName</i>	The name of an existing hotel Note: To be used along with <i>hotelAddress</i>	String representing the name of a hotel
** <i>hotelAddress</i>	The address of an existing hotel Note: To be used along with <i>hotelName</i>	String representing the address of a hotel
* <i>contactPerson</i>	The surname and last name of the contact person	String representing a surname and last name
* <i>contactCellphone</i>	The contact cellphone number	String representing a cell phone number
* <i>airlineCruiseline</i>	The name of the airline/cruiseline used for the arrival transfer.	String representing the name of an airline/cruiseline

<i>*flightNumber</i>	The flight number used for the arrival transfer	String representing a flight number
<i>*originTerminal</i>	The name of the origin terminal, required to book cruise transfers.	String representing the name of the terminal of origin
<i>*departureIataCode</i>	The IATA code of the origin airport, required to book arrival transfers.	A string representing the departure airport IATA Code. Must be a valid IATA airport code
<i>*departureTime</i>	Time of departure	A timestamp in the format "hh:mm", where hh = hours and mm = minutes
<i>*arrivalTime</i>	Time of arrival	A timestamp in the format "hh:mm", where hh = hours and mm = minutes
<i>*arrivalDate</i>	Date of arrival	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number
<i>*passengers</i>	The number of passengers to transfer	Integers > 0
<i>*transferID</i>	The ID of the associated transfer	Integer representing the ID of a transfer
<i>*returnTransfer</i>	Indicates whether a return transfer should be booked	0 = only arrival transfer or 1 = both arrival and return transfer or 2 = only return transfer
<i>*returnAirlineCruiseline</i>	The name of the airline/cruiseline used for the return transfer Note: This is only required if "returnTransfer" is set to 1	String representing the name of an airline/cruiseline
<i>*returnFlightNumber</i>	The flight number used for the return transfer Note: This is only required if "returnTransfer" is set to 1	String representing a flight number
<i>*returnDepartureDate</i>	Date of departure Note: This is only required if "returnTransfer" is set to 1	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number
<i>*returnDepartureTime</i>	Time of departure Note: This is only required if "returnTransfer" is set to 1	A timestamp in the format "hh:mm", where hh = hours and mm = minutes
<i>invoiceRef</i>	An invoice reference.	A string representing your invoice reference number.

<i>yourRef</i>	A booking number or reference.	A string representing your reference.
----------------	--------------------------------	---------------------------------------

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx?op=AddTransferV2>

XML Output

```
<?xml version="1.0" encoding="utf-8"?>
<TransferBookingList>
  <!--This list will only contain information about the transfer that has been
booked.-->
  <transferbookings>
    <transferbooking>
      <!--The ID of the booking where the transfer was added.-->
      <transferbookingnumber></transferbookingnumber>
      <!--The ID of the hotel where the transfer is booked-->
      <hotel.id></hotel.id>
      <!--The name of the hotel where the transfer is booked.-->
      <hotel.name></hotel.name>
      <!--The name of the contact person for the transfer booking.-->
      <contactPerson></contactPerson>
      <!--The contact cellphone number for the transfer booking.-->
      <contactCellphone></contactCellphone>
      <!--The airline used.-->
      <airline></airline>
      <!--The flight number used.-->
      <flightNumber></flightNumber>
      <!--The airport used for departure.-->
      <departureAirport></departureAirport>
      <!--The airport used for arrival.-->
      <arrivalAirport></arrivalAirport>
      <!--The time of departure (in the format hh:mm, where hh = hours,
ranging from 00 to 23, mm = minutes, ranging from 00 to 59).-->
      <departureTime></departureTime>
      <!--The date of arrival (in the format YYYY-MM-DD,
where YYYY = year, MM = month number and DD = day number).-->
      <arrivalDate></arrivalDate>
      <!--The time of arrival (in the format hh:mm, where hh = hours,
ranging from 00 to 23, mm = minutes, ranging from 00 to 59).-->
      <arrivalTime></arrivalTime>
      <!--The number of passengers.-->
      <passengers></passengers>
      <!--The ID of the associated transfer.-->
      <transferID></transferID>
      <!--The ID of the transfer type.-->
      <transferTypeID></transferTypeID>
      <!--The name of the transfer type.-->
      <transferType></transferType>
      <!--Text that is shown on the arrival voucher.-->
      <arrivalVoucherText></arrivalVoucherText>
      <!--Text that is shown on the return voucher.-->
      <returnVoucherText></returnVoucherText>
      <!--The URL for the voucher.-->
      <voucherURL></voucherURL>
      <!--The provider from the transfer voucher.-->
      <provider></provider>
      <!--The phone numbers from the transfer voucher.-->
      <providerPhone></providerPhone>
      <!--The emergency phone numbers from the transfer voucher.-->
      <providerEmergencyPhone></providerEmergencyPhone>
```

```

<!--Integer indicating whether a return transfer is included.
1 = return transfer included, 0 = return transfer not included.-->
<returntransfer></returntransfer>
<!--The airline used for the return transfer.-->
<returnairline></returnairline>
<!--The flight number used for the return transfer.-->
<returnflightNumber></returnflightNumber>
<!--The date of departure for the return transfer.-->
<returndepartureDate></returndepartureDate>
<!--The time of departure for the return transfer.-->
<returndepartureTime></returndepartureTime>
<!--The price for the transfer in the used currency.-->
<price></price>
<!--The used currency.-->
<currency></currency>
<!--The price for the transfer in Euros.-->
<price.EUR></price.EUR>
<!--The status of the booking.-->
<bookingstatus>NotActive or Active</bookingstatus>
<!--The invoice reference used when booking.-->
<invoiceref></invoiceref>
<!-- The reference used when booking.-->
<yourref></yourref>
</transferbooking>
</transferbookings>
<!--This element only appears when an error has occurred.-->
<Error>
  <!--The type of error (the exception type).-->
  <ErrorType></ErrorType>
  <!--A description of the error.-->
  <Message></Message>
</Error>
</TransferBookingList>

```

Errors

Listed below are common errors that may appear under the “Error” element. Should any errors occur that are not listed here, contact Sunhotels.

Error Type	Description	Resolution
InvalidParametersForTransferException	You are trying to book an arrival-only transfer but have provided return-specific parameters to the function.	Remove the values for the return-specific parameters.
NoTransfersAvailableForCCAgentsException	You are trying to book a transfer while the “force_creditcard” setting for your account is set to “1”, which is not possible.	N/A
NonExistentTransferException	You are trying to book a transfer that does not exist.	Use the “SearchTransfers” function to find available transfers to book.

BookingNotActiveException	You are trying to add a transfer to a hotel booking that is not active for some reason (e.g. have been cancelled, etc.)	Add the transfer to a different booking or as a stand-alone transfer booking.
PassengersLessThanTransferMinimumException	The provided number of passengers is less than the minimum allowed number of passengers for the transfer.	Increase the number of passengers to meet the minimum criterion, or try to book a different transfer.
PassengersMoreThanTransferMaximumException	The provided number of passengers is more than the maximum allowed number of passengers for the transfer.	Decrease the number of passengers to meet the minimum criterion, or try to book a different transfer.
TransferAlreadyBookedException	You are trying to add a transfer to a booking that already has a transfer added to it.	Add the transfer to a different booking or cancel the existing transfer booking.

Cancelling a transfer booking

This function is used to cancel an existing transfer booking.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/CancelTransferBooking?userName=\[YourUsername\]&password=\[YourPassword\]&language=en&email=noreply@sunhotels.net&transferBookingID=ST8000506](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/CancelTransferBooking?userName=[YourUsername]&password=[YourPassword]&language=en&email=noreply@sunhotels.net&transferBookingID=ST8000506)

Function inputs

*: Required input

Parameter name	Description	Valid Input Values
* <i>userName</i>	The name of the user that is performing the cancellation operation.	Non-empty strings
* <i>password</i>	The password of the user that is performing the cancellation operation.	Non-empty strings
* <i>language</i>	Determines the language that the booking cancellation mail will be displayed in.	A valid ISO code from the " GetLanguages " function.
* <i>email</i>	The e-mail address of the agent. This is where the cancellation e-mail is sent.	String representing a valid e-mail address.
* <i>transferBookingID</i>	The ID of the booking whose transfer is to be cancelled.	String containing numerical characters, optionally preceded by "ST"

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.aspx?op=CancelTransferBooking>

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<cancelTransferBookingResult>
  <!--The generated cancellation fee (if the cancellation yielded a fee,
  otherwise 0), in Euros.-->
  <cancellationFeeInEuros></cancellationFeeInEuros>
  <!--The generated cancellation fee (if the cancellation yielded a fee,
  otherwise 0), in currency specified in the booking.-->
  <cancellationFeeInLocalCurrency></cancellationFeeInLocalCurrency>
  <!--The currency specified in the cancelled transfer booking.-->
  <currency></currency>
  <!--The status code of the cancellation.
  1 is successful and -1 is unsuccessful.-->
  <code></code>
  <!--This element only appears when an error has occurred.-->
  <Error>
    <!--The type of error (exception).-->
    <ErrorType></ErrorType>
    <!--A description of the error.-->
    <Message></Message>
  </Error>
</cancelTransferBookingResult>
```

Errors

Listed below are common errors that may appear under the “Error” element.

Should any errors occur that are not listed here, contact Sunhotels.

Error Type	Description	Resolution
NonExistentTransferBookingException	You are trying to cancel a transfer booking for a booking that have not had any transfers added to it.	Check that the provided booking ID is correct.
NonExistentTransferBookingException	You are trying to cancel a transfer booking that has already been cancelled.	Not available
TransferCancellationDeadlineExpiredException	The cancellation deadline of the transfer booking has expired.	Not available.

Downloading static data

This function is used to download exported XML files containing static information about hotels, destinations, etc. in the system.

Example URL with input:

[http://xml.sunhotels.net/15/DownloadExport.aspx?username=\[YourUsername\]&password=\[YourPassword\]&language=en&xmlType=Place&version=150&Format=zip](http://xml.sunhotels.net/15/DownloadExport.aspx?username=[YourUsername]&password=[YourPassword]&language=en&xmlType=Place&version=150&Format=zip)

Function inputs

*: Required input

**: Required alternative (one of these parameters must be specified)

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>xmlType</i>	The type of contents of interest. "Place" indicates information about destinations, countries, cities, airports, etc. "Cache" indicates information about things such as hotels, rooms, etc.	"Place" or "Cache"
* <i>language</i>	The language in which the results will be displayed. Note: Contact Sunhotels to know what languages are available on your account.	A valid ISO code from the "GetLanguages" function.
<i>version</i>	The version of the XML content. 150 have to be specified to get the most current resort mapping. The value 150 stands for the schema 1.5.0. Newer versions will be released in the future.	150
<i>Format</i>	Format to compress the download of the file.	"zip" is the compression format used

Content of places document:

The file consist of nested places elements. The first level of places elements should be interpreted as countries. The next level of places should be interpreted as destinations. The third and most inner level is resorts.

Example of a places document:

```
<?xml version="1.0" encoding="utf-8"?>
<sunhotels_cache_file xmlns="http://xml.sunhotels.net/schemas/Places_1.5.0.xsd">
  <provider_definition>
    <name>InternalExport</name>
    <currency>EUR</currency>
  </provider_definition>
  <code_types>
    <code_type>IATA</code_type>
    <code_type>ISO-COUNTRY</code_type>
  </code_types>
  <places>
    <!--First level: Countries-->
    <place>
      <id>1</id>
      <description>Spain</description>
      <codes>
        <code type="ISO-COUNTRY" value="ES" />
      </codes>
      <places>
        <!--Second level: Destinations-->
        <place>
          <id>4</id>
          <description>Mallorca</description>
          <codes>
            <code type="IATA" value="PMI" />
          </codes>
          <places>
            <!--Third level: Resorts-->
            <place>
              <id>9</id>
              <description>Cala d Or</description>
            </place>
            <place>
              <id>22</id>
              <description>Alcudia Bay</description>
            </place>
            <place>
              <id>30</id>
              <description>Cala Vinas</description>
            </place>
          </places>
        </place>
      </places>
    </place>
  </places>
</sunhotels_cache_file>
```

Content of cache document

The cache document contains all static data about a hotel, like the name, address and description. The cache file consists of two main elements, hotels and room types. All hotels also have one or more rooms. A room must have a room type that can be found under the room types definitions.

Example of cache document:

```
<?xml version="1.0" encoding="utf-8"?>
<sunhotels_cache_file
xmlns="http://xml.sunhotels.net/schemas/AvailabilityCache_1.5.0.xsd">
  <provider_definition>
    <name>InternalExport</name>
    <currency>EUR</currency>
  </provider_definition>
  <passenger_definitions>
    <child max_age="11" />
  </passenger_definitions>
  <features>
    <feature>airconditioning</feature>
    <feature>elevator</feature>
    <feature>bar</feature>
    <feature>telephone</feature>
    <feature>childrens_pool</feature>
    <feature>pool</feature>
    <feature>safe</feature>
    <feature>sea_view</feature>
    <feature>balcony</feature>
    <feature>restaurant</feature>
    <feature>tv</feature>
    <feature>wireless internet</feature>
  </features>
  <themes>
    <theme>luxury</theme>
    <theme>budget</theme>
    <theme>spa</theme>
    <theme>family</theme>
    <theme>skiing</theme>
    <theme>eco-certified</theme>
  </themes>
  <distance_types>
    <distance_type>center</distance_type>
    <distance_type>airport</distance_type>
    <distance_type>beach</distance_type>
  </distance_types>
  <!--Room types will be used inside the room on each hotel.-->
  <room_types>
    <type>
      <description>Apartment 1 bedroom - Sea View</description>
      <id>51.4.0</id>
      <beds>4</beds>
      <extrabeds>0</extrabeds>
      <shared_room>false</shared_room>
      <shared_facilities>false</shared_facilities>
    </type>
    <type>
      <description>Apartment 1 bedroom - Pool View</description>
      <id>52.4.0</id>
      <beds>4</beds>
      <extrabeds>0</extrabeds>
      <shared_room>false</shared_room>
      <shared_facilities>false</shared_facilities>
    </type>
  </room_types>
</sunhotels_cache_file>
```

```
<Cancellation_policies />
<hotels>
```

```
<hotel>
```

```
<id>535</id>
```

```
<name>Niu Daus</name>
```

```
<accommodation_type>apartment</accommodation_type>
```

```
<headline>Apartments close to Porto Petro (700 m).</headline>
```

```
<description>
```

Niu D'Aus is a small apartment complex situated on the road to Porto Petro, just 1,5 km outside the centre of Cala Egos, where you will find the nearest beach and shops, bars and restaurants.

The apartments consist of a twin bedroom, living room with sofa beds, kitchenette, bathroom and balcony. There is a swimming pool with a children's section, small playground, sun terrace.

There is a tennis court 5 min from the hotel and just 4 km away there are 8 different beaches.

Cleaning services and change of towels: 3 times weekly and change of sheets: once a week.

Reception with limited opening hours. Important - While making the booking you are requested to indicate the arrival time.

```
</description>
```

```
<place>9</place>
```

```
<address>
```

```
<street1>S'Estel d'Derjen 18</street1>
```

```
<street2>(carretera Porto Petro Cala d'or)</street2>
```

```
<zipcode>07691</zipcode>
```

```
<city>Porto Petro</city>
```

```
<state>Mallorca</state>
```

```
<country>Spain</country>
```

```
</address>
```

```
<position>
```

```
<latitude>39.3638830313957</latitude>
```

```
<longitude>3.21358680725098</longitude>
```

```
</position>
```

```
<phone>0034 971 658 114</phone>
```

```
<fax>0034 971 659 767</fax>
```

```
<email>
```

```
</email>
```

```
<classification>3</classification>
```

```
<best_buy>true</best_buy>
```

```
<!-- If set to 1 minimum age to stay at hotel is 18 years -->
```

```
<adult_only>1</adult_only>
```

```
<features>
```

```
<feature>airconditioning</feature>
```

```
<feature>childrens_pool</feature>
```

```
<feature>pool</feature>
```

```
<feature>safe</feature>
```

```
<feature>balcony</feature>
```

```
<feature>tv</feature>
```

```
<feature>wireless internet</feature>
```

```
</features>
```

```
<distances>
```

```
<distance name="center" value="0.7" />
```

```
<distance name="airport" value="60" />
```

```
<distance name="beach" value="1.5" />
```

```
</distances>
```

```
<reviews>
```

```
<review provider="CHR" type="ReviewAvgRating" value="3.67" />
```

```
<review provider="CHR" type="ReviewAvgPriceValue" value="4.00" />
```

```
<review provider="CHR" type="ReviewCount" value="12" />
```

```

        <review provider="CHR" type="ReviewRecommendCount" value="0" />
    </reviews>
    <images>
        <image id="10473272">
            <image_variant width="200"
height="150"> https://hotelimages.sunhotels.net/HotelInfo/hotelImage.aspx?id=10473272
        </image_variant>
            <image_variant width="800"
Height="600"> https://hotelimages.sunhotels.net/HotelInfo/hotelImage.aspx?full=1&a
mp;id=10473272</image_variant>
        </image>
        <image id="10473272">
            <image_variant width="200"
height="150">https://hotelimages.sunhotels.net/HotelInfo/hotelImage.aspx?id=10473272
        </image_variant>
            <image_variant width="800"
height="600">https://hotelimages.sunhotels.net/HotelInfo/hotelImage.aspx?full=1&a
mp;id=10473272</image_variant>
        </image>
        <image id="10473272">
            <image_variant width="200"
height="150">https://hotelimages.sunhotels.net/HotelInfo/hotelImage.aspx?id=10473272
        </image_variant>
            <image_variant width="800"
height="600">https://hotelimages.sunhotels.net/HotelInfo/hotelImage.aspx?full=1&a
mp;id=10473272</image_variant>
        </image>
    </images>
    <rooms>
        <!--The definition of the type_id attribute is in room types-->
        <room type_id="51.4.0">
            room_id>715</room_id>

        </room>
        <room type_id="52.4.0">
            <room_id>15972</room_id>
            <best_buy>>false</best_buy>
        </room>
    </rooms>
</hotel>
</hotels>
</sunhotels_cache_file>

```

Country, Destination and Resort notes

City tax information and other important notes related to some countries, destinations or resorts, like nationality restrictions, are only returned in PreBook and booking response in our API. If you wish to also show this information in search you can retrieve the information of which notes exist and what days they apply on by downloading following files.

*Make sure to update below URLs to use your username, password and language.

Example URL with input:

Country Notes

[http://xml.sunhotels.net/15/DownloadExport.aspx?username=\[YourUsername\]&password=\[YourPassword\]&language=XX&xmlType=country_notes&version=150](http://xml.sunhotels.net/15/DownloadExport.aspx?username=[YourUsername]&password=[YourPassword]&language=XX&xmlType=country_notes&version=150)

Destinations Notes

[http://xml.sunhotels.net/15/DownloadExport.aspx?username=\[YourUsername\]&password=\[YourPassword\]&language=XX&xmlType=destination_notes&version=150](http://xml.sunhotels.net/15/DownloadExport.aspx?username=[YourUsername]&password=[YourPassword]&language=XX&xmlType=destination_notes&version=150)

Resort Notes

[http://xml.sunhotels.net/15/DownloadExport.aspx?username=\[YourUsername\]&password=\[YourPassword\]&language=XX&xmlType=resort_notes&version=150](http://xml.sunhotels.net/15/DownloadExport.aspx?username=[YourUsername]&password=[YourPassword]&language=XX&xmlType=resort_notes&version=150)

Retrieving all supported languages

This function is used to retrieve a list of all supported languages, including shortened versions of their ISO codes that are to be used in all other function with a “language” parameter. Note that before getting access to the language it need to be enabled on your account. Speak to your sales representative at Sunhotels for more information.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetLanguages?userName=\[YourUsername\]&password=\[YourPassword\]](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetLanguages?userName=[YourUsername]&password=[YourPassword])

Function inputs

*: Required input

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx?op=GetLanguages>

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<getLanguagesResult>
  <!-- All retrieved languages-->
  <languages>
    <!--isoCode: The code to use in the "language" parameter in all functions
         that have the parameter, i.e. "en", "fr", etc.
         name: The english name of the language, i.e. "English", "French", etc. -->
    <language isoCode="" name="" />
  </languages>
</getLanguagesResult>
```


Retrieving all meals

This function is used to retrieve all meals in the system, along with their various labels, translated to a specified language.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetMeals?userName=\[YourUsername\]&password=\[YourPassword\]&language=en](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetMeals?userName=[YourUsername]&password=[YourPassword]&language=en)

Function inputs

*: Required input

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>language</i>	This string will determine the language that the meals and labels will be displayed in.	A valid ISO code from the "GetLanguages" function.

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx?op=GetMeals>

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<getMealsResult>
  <!--All available meals-->
  <meals>
    <meal>
      <!--The ID of the meal.-->
      <id></id>
      <!--The translated name of the meal.-->
      <name></name>
      <!--All labels for the meal-->
      <labels>
        <label>
          <!--The ID of the meal label.-->
          <id></id>
          <!--The text of the meal label.-->
          <text></text>
        </label>
      </labels>
    </meal>
  </meals>
  <!--This element only appears when an error has occurred.-->
  <Error>
    <!--The type of error (the exception type).-->
    <ErrorType></ErrorType>
    <!--A description of the error.-->
    <Message></Message>
  </Error>
</getMealsResult>
```

Retrieving all features

This function is used to retrieve all features in the system, along with their various labels, translated to a specified language.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetFeatures?userName=\[YourUsername\]&password=\[YourPassword\]&language=en](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetFeatures?userName=[YourUsername]&password=[YourPassword]&language=en)

Function inputs

*: Required input

Parameter name	Description	Valid Input Values
* <i>username</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>language</i>	This string will determine the language that the features will be displayed in.	A valid ISO code from the “GetLanguages” function.

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx?op=GetFeatures>

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<getFeaturesResult>
  <features>
    <!--id: The ID of the feature, name: The translated name of the feature-->
    <feature id="" name="" />
  </features>
</getFeaturesResult>
```

Retrieving all hotel note descriptions

This function is used to retrieve all hotel note descriptions in the system, translated to a specified language.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetHotelNoteTypes?userName=\[YourUsername\]&password=\[YourPassword\]&language=en](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetHotelNoteTypes?userName=[YourUsername]&password=[YourPassword]&language=en)

Function inputs

*: Required input

Parameter name	Description	Valid Input Values
* <i>username</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>language</i>	This string will determine the language that the notes will be displayed in.	A valid ISO code from the “GetLanguages” function.

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx?op=GetHotelNoteTypes>

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<getHotelNoteTypesResult>
  <noteTypes>
    <!--id: The ID of the note type, text: The translated text of the note type-->
    <noteType id="" text="" />
  </noteTypes>
</getHotelNoteTypesResult>
```

Retrieving all room note descriptions

This function is used to retrieve all room note descriptions in the system, translated to a specified language.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetRoomNoteTypes?userName=\[YourUsername\]&password=\[YourPassword\]&language=en](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetRoomNoteTypes?userName=[YourUsername]&password=[YourPassword]&language=en)

Function inputs

*: Required input

Parameter name	Description	Valid Input Values
* <i>username</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>language</i>	This string will determine the language that the notes will be displayed in.	A valid ISO code from the “GetLanguages” function.

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx?op=GetRoomNoteTypes>

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<getRoomNoteTypesResult>
  <noteTypes>
    <!--id: The ID of the note type, text: The translated text of the note type-->
    <noteType id="" text="" />
  </noteTypes>
</getRoomNoteTypesResult>
```

Retrieving booking special request

This function is used to retrieve a booking special request from the system.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetSpecialRequest?userName=\[YourUsername\]&password=\[YourPassword\]&bookingId=2435578](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/GetSpecialRequest?userName=[YourUsername]&password=[YourPassword]&bookingId=2435578)

Function inputs

*: Required input

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>bookingId</i>	This booking ID of the booking to get the special request for.	Integers > 0

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx?op=GetSpecialRequest>

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<getSpecialRequestResult>
  <!--text: The text describing the special request, empty if no special
    request is available-->
  <text />
</getSpecialRequestResult>
```

Updating booking special request

This function is used to update a booking special request.

Example URL with input:

[http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/UpdateSpecialRequest?userName=\[YourUsername\]&password=\[YourPassword\]&bookingId=2435578&text=This%20is%20a%20special%20request](http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx/UpdateSpecialRequest?userName=[YourUsername]&password=[YourPassword]&bookingId=2435578&text=This%20is%20a%20special%20request)

Function inputs

*: Required input

Parameter name	Description	Valid Input Values
* <i>userName</i>	The user name of the agent that is calling the function.	Non-empty strings
* <i>password</i>	The password of the agent that is calling the function.	Non-empty strings
* <i>bookingId</i>	This booking ID of the booking to get the special request for.	Integers > 0
<i>text</i>	The special request text.	strings

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/15/PostGet/NonStaticXMLAPI.asmx?op=UpdateSpecialRequest>

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<updateSpecialRequestResult>
  <!--Declares whether the update was successful.
    0: Booking does not exist, 1: Updated, 2: No changes were done.-->
  <resultCode />
  <!--A message describing the result code.-->
  <resultMessage />
</updateSpecialRequestResult>
```