

XML API, version 14

Release 2011:02

Document version: 2014-09-15

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).

Table of contents

Changelog	3
Introduction	6
Global errors	7
Searching.....	9
Retrieving room type list [deprecated]	20
Booking.....	22
Cancelling a booking	29
Retrieving a list of transfer types	31
Retrieving a list of themes	32
Retrieving a list of destinations [deprecated].....	33
Retrieving a list of resorts [deprecated]	35
Transfer searching.....	37
Retrieving a list of static hotels and rooms.....	40
Retrieving booking information	45
Retrieving transfer booking information.....	49
Booking a transfer	52
Cancelling a transfer booking	57
Downloading static data	59
Retrieving all supported languages	64
Retrieving all meals	65
Retrieving all features	66

Changelog

This section describes all changes made to the API since v13, function by function.

New functions

The function [“GetLanguages”](#) has been added, which is used for retrieving all supported languages.

The function [“GetMeals”](#) has been added, which is used for retrieving all meals, along with their labels.

The function [“GetFeatures”](#) has been added, which is used to retrieve all features.

Global changes

The *“language”* parameter in all functions that require it is now a string, whose valid values can be retrieved by calling the new [“GetLanguages”](#) function.

Search

The parameters *“noMeals”*, *“breakfast”*, *“halfBoard”*, *“fullBoard”* and *“allInclusive”* have been replaced with the *“mealIds”* parameter.

The parameter *“showTransfer”* has been removed.

The following function inputs have been added: *showReviews*, *referencePointLatitude*, *referencePointLongitude* and *maxDistanceFromReferencePoint*, *minStarRating*, *maxStarRating*, *featureIds*, *minPrice*, *maxPrice*, *themelds*, *excludeSharedRooms* and *excludeSharedFacilities* and *prioritizedHotelId*.

The following changes have been made to the output:

- Non-static searching
 - Removed *“special.flag”* and *“transfers”* elements from hotels.
 - Added *“review”*, *“notes”* and *“isPrioritized”* elements to hotels
 - Removed *“price.EUR”*, *“price”* and *“special.offer”* elements from rooms
 - Added *“notes”*, *“isSuperDeal”* and *“isBestBuy”* elements to rooms.
 - Removed *“name”* element from meals.
 - Added *“id”*, *“labelId”* and *“discount”* elements to meals.
- Static searching
 - All changes made to the non-static search result are mirrored in the static search result
 - The *“name”* element under all meals now contains the translated name of the meal.
 - Added *“labelText”* element to meals.
 - Added *“features”* and *“themes”* elements to hotels and rooms.
 - Added *“sharedRoom”* and *“sharedFacilities”* elements to room types.
 - Removed *“imageUrls”* element from hotels (replaced with *“images”*)
 - Removed the *“dist.beach”*, *“dist.centre”* and *“dist.airport”* elements from hotels (replaced with *“distanceTypes”*).
 - Added the following elements to hotels: *“hotel.addr.countrycode”*, *“themes”*, *“images”*, *“features”*, *“distanceTypes”*, *“timeZone”* and *“isBestBuy”*.
- **MarketSearch**

New endpoints available with updated search and book function including:

- A *“customerCountry”* parameter has been added, where the nationality of the customer is specified.
- A *“B2C”* parameter has been added, which specifies if the client comes from a B2B/Package or B2C point of sale.

[See correct URLs to use for Search with Market and B2C parameters](#)

GetRoomTypes

The *“RoomTypeList”* root element has been renamed to *“getRoomTypesResult”*.

The *“RoomTypes”* element was renamed to *“roomTypes”*.

The *“RoomType”* element was renamed to *“roomType”*.

The “*ID*” element under each room type was renamed to “*id*”, “*Name*” was renamed to “*name*” and “*Shared*” was renamed to “*sharedRoom*”.

The “*sharedFacilities*” element was added to each room type.

Book

The root element in the XML output was renamed to “*bookResult*”.

The first and last names of all adult and child guests must now be entered separately

The “*meal*” parameter was replaced with the “*mealId*” parameter.

The same XML output is now yielded for this function as for the “*GetBookingInformation*” function.

Added the following elements to the XML output (the same elements were also added to the XML output of “*GetBookingInformation*”):

“*hotel.id*”, “*room.englishType*”, “*englishBoard*”, “*earliestNonFreeCancellationDate.CET*”, “*earliestNonFreeCancellationDate.Local*”, “*hotelNotes*”, “*englishHotelNotes*”, “*roomNotes*” and “*englishRoomNotes*”.

- **MarketSearch**

New endpoints available with updated search and book function including:

- A “*customerCountry*” parameter has been added, where the nationality of the customer is specified.
- A “B2C” parameter has been added, which specifies if the client comes from a B2B/Package or B2C point of sale.

[See correct URLs to use for Book with Market and B2C parameters](#)

GetFlagTypes

This function has been renamed to “*GetThemes*”, and the name of the root element is now “*getThemesResult*”.

The input parameter “*flagCode*” was removed from the function. All themes are now always returned.

The “*Id*” element under each flag type was renamed to “*id*”, “*Name*” was renamed to “*name*” and

“*Image*” was renamed to “*image*”.

GetDestinations

The “*CountryCode*” element was added to each destination.

GetResorts

The “*CountryCode*” element was added to each resort.

SearchTransfers

The casing for all elements in the XML output has been corrected in accordance with XML naming standards.

The root element “*TransferList*” was renamed to “*searchTransfersResult*”.

The following function inputs have been added:

“*arrivalDate*”, “*arrivalTime*”, “*returnDepartureDate*” and “*returnDepartureTime*”.

The following elements have been added to each airport: “*airportId*”, “*resortId*”, “*airportName*”, “*rates*” and “*timeZone*”.

The “*IATA*” element under each airport was renamed to “*iataCode*”.

The “*transferTypeName*” element was added to each transfer.

The “*ID*” element for each transfer element was renamed to “*transferId*”, “*minpassengers*” was

renamed to “*minimumPassengers*”, “*maxpassengers*” was renamed to “*maximumPassengers*”,

“*estimatedtimeHH*” was renamed to “*estimatedTimeInHours*” and “*estimatedtimeMM*” was renamed to “*estimatedTimeInMinutes*”.

The following elements were removed from each transfer element: “*priceEUR*”, “*price*” and “*allowReturn*”.

SearchRooms

This function was renamed to “*GetStaticHotelsAndRooms*”.

The changes made to the XML output in the “*Search*” function is mirrored in this function.

The “*accommodationType*” parameter has been renamed to “*accommodationTypes*”, which now accepts a list of accommodation types.

GetBookingInformation

The root element in the XML output was renamed to `getBookingInformationResult`.

The same XML output is now yielded (except for the `bookingStatus` element, which is not yielded when creating a new booking) for each booking in this result set as when using the `Book` function.

The `Bookings` element under the `BookingList` element was renamed to `bookings`.

GetTransferBookingInformation

The root element in the XML output was renamed to `getTransferBookingInformationResult`.

The `voucherText` element was replaced with `arrivalVoucherText` and `returnVoucherText`

The following elements were added to each transfer booking:

`hotel.id`, `voucherURL`, `provider`, `providerPhone` and `providerEmergencyPhone`.

The `transferbookings` element was renamed to `transferBookings` and each `transferbooking` element was renamed to `transferBooking`.

AddTransfer

The XML output when booking a transfer is now identical to the XML output of a transfer booking when using the `GetTransferBookingInformation` function, except for the root element, which is named `addTransferResult`.

CancelTransferBooking

The root `TransferBookingCancellationResult` element was renamed to

`cancelTransferBookingResult`, `CancellationFeeInEuros` was renamed to `cancellationFeeInEuros`,

`CancellationFeeInLocalCurrency` was renamed to `cancellationFeeInLocalCurrency`, `Currency` was renamed to `currency` and `Code` was renamed to `code`.

The existing `Code` element in the XML output is now included and described in the documentation.

GetTransferTypes

The `TransferTypeList` element was renamed to `getTransferTypesResult`, `TransferTypes` was renamed to `transferTypes` and each `TransferType` element was renamed to `transferType`.

The `ID` element for each transfer type was renamed to `id` and `Name` was renamed to `name`.

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.

Two new endpoints have been added including search and book functions with *customerCountry* and *B2C* parameters, replacing the search and book functions in the previous web services.

	POST / GET	SOAP
Show static data	/14/PostGet/StaticXMLAPI.asmx	/14/SOAP/StaticXMLAPI.asmx
Do not show static data	/14/PostGet/NonStaticXMLAPI.asmx	/14/SOAP/NonStaticXMLAPI.asmx
Show static data - Market and B2C	/14/PostGet/StaticMarketSearch.asmx	/14/SOAP/StaticMarketSearch.asmx
Do not show static data -Market and B2C	/14/PostGet/NonStaticMarketSearch.asmx	/14/SOAP/NonStaticMarketSearch.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/14/PostGet/NonStaticMarketSearch.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!

2 new endpoints have been added including search and book function and should be used from now. These new search and book functions include 2 new parameters, *customerCountry* and *B2C*, that will allow to dynamically advise client nationality and point of sale in search and booking request to SunHotels.

[See correct URLs to use for Search with Market and B2C parameters](#)

Non-static searching

Example URL with input for non-static results:

<http://xml.sunhotels.net/14/PostGet/NonStaticMarketSearch.aspx/Search?userName=testagent&password=785623&language=en¤cy=SEK&checkInDate=2011-08-02&checkOutDate=2011-08-04&numberOfRooms=1&destination=&destinationID=695&hotelIDs=&resortIDs=&accommodationTypes=&numberOfAdults=2&numberOfChildren=0&sortBy=&sortOrder=&exactDestinationMatch=&blockSuperdeal=&showTransfer=&mealIDs=&showCoordinates=&showReviews=&referencePointLatitude=&referencePointLongitude=&maxDistanceFromReferencePoint=&minStarRating=&maxStarRating=&featureIDs=&minPrice=&maxPrice=&themeIDs=&excludeSharedRooms=&excludeSharedFacilities=&prioritizedHotelIDs=&customerCountry=gb&B2C=>

Static searching

Example URL with input for static results:

<http://xml.sunhotels.net/14/PostGet/StaticMarketSearch.aspx/Search?userName=testagent&password=785623&language=en¤cy=SEK&checkInDate=2011-08-02&checkOutDate=2011-08-04&numberOfRooms=1&destination=&destinationID=695&hotelIDs=&resortIDs=&accommodationTypes=&numberOfAdults=2&numberOfChildren=0&sortBy=&sortOrder=&exactDestinationMatch=&blockSuperdeal=&showTransfer=&mealIDs=&showCoordinates=&showReviews=&referencePointLatitude=&referencePointLongitude=&maxDistanceFromReferencePoint=&minStarRating=&maxStarRating=&featureIDs=&minPrice=&maxPrice=&themeIDs=&excludeSharedRooms=&excludeSharedFacilities=&prioritizedHotelIDs=&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>currency</i>	The currency that all prices will be converted to in the search results.	String representing a currency code, e.g. "GBP".
* <i>checkInDate</i>	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.	A date in the format 'YYYY-MM-DD' YYYY = Year MM = Month number DD = Day number

<i>*checkOutDate</i>	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.	A date in the format 'YYYY-MM-DD', where YYYY = Year MM = Month number DD = Day number
<i>*numberOfRooms</i>	The number of rooms that the client is interested in.	An integer value between 1 and the number of adults specified in " <i>numberOfAdults</i> "
<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.	A string containing hotel IDs, separated by commas. Example: "12,34,56"
<i>**resortIDs</i>	The ID(s) of the resort(s) where the client wishes to look for rooms.	A string containing resort IDs, separated by commas. Example: "12,34,56"
<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>	The number of children the client wants the rooms to be able to accommodate. Note: Child is between 2 and 11 years old. 0-1 years old is infant (not used in search), above 11 years is adult.	An integer value between 0 and 9
<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 reach 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>	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). Note: If provided, the <i>referencePointLatitude</i> and <i>maxDistanceFromReferencePoint</i> parameters must also be provided.	A decimal value >= -180 and <= 180

<i>maxDistanceFromReferencePoint</i>	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>) Note: If provided, the <i>referencePointLatitude</i> and <i>referencePointLongitude</i> parameters must also be provided.	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>	The IDs of the features that the hotels and rooms should have. Note: To retrieve the IDs to use for this parameter, use the new "GetFeatures" function.	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>themelds</i>	The IDs of the themes that the hotels must have. Note: To get available theme IDs, use the "GetThemes" function.	A string containing the theme IDs, separated by commas, e.g. "1,2"
<i>excludeSharedRooms</i>	Whether or not to exclude shared rooms.	"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>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".

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/14/PostGet/NonStaticMarketSearch.aspx?op=Search>

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<searchresult>
  <!--All retrieved hotels-->
  <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 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>
          <!--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 beds that are available for the room-->
              <beds></beds>
              <!--The number of extra beds 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 price for the room and the meal (in Euros)-->
                  <price_in_EUR></price_in_EUR>
                  <!--The combined price for the room and the meal (in the currency
                  specified for the search)-->
                  <price></price>
                  <!--The discount that is included in the price.-->
                  <discount>
                    <!--The discount in Euros-->
                    <discountInEuros></discountInEuros>
                    <!--The discount in the currency specified for the search-->
                    <discountInLocalCurrency></discountInLocalCurrency>
                    <!--The ID of the discount type-->
```

```

        <typeId></typeId>
    </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 a booking for the room.-->
        <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>
</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 />
<!--Whether or not the hotel is prioritized in the results-->
<isPrioritized></isPrioritized>
</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/14/PostGet/StaticMarketSearch.aspx?op=Search>

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<searchresult>
  <hotels>
    <hotel xsi:type="StaticHotel">
      <!--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 beds that are available for the room-->
              <beds></beds>
              <!--The number of extra beds 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 price for the room and the meal (in Euros)-->
                  <price_in_EUR></price_in_EUR>
                  <!--
The combined price for the room and the meal (in the currency
specified for the search)-->
                  <price></price>
                  <!--The discount that is included in the price.-->
                  <discount>
                    <!--The discount in Euros-->
                    <discountInEuros></discountInEuros>
                    <!--The discount in the currency specified for the search-->
                    <discountInLocalCurrency></discountInLocalCurrency>
                    <!--The ID of the discount type-->
```

```

        <typeId></typeId>
    </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 a booking for the room.-->
        <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 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-->
<sharedRoom></sharedRoom>

```



```

        <!--Whether facilities are shared for this room type-->
        <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.
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 or not the hotel is prioritized in the results-->
<isPrioritized></isPrioritized>
<!--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 "http://www.sunhotels.net/SunHotels.net/Hot
lInfo/hotelImage.aspx ?id=239846&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 "http://www.sunhotels.net/SunHotels.net/Hot
lInfo/hotelImage.aspx ?id=239846")
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>
</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”

Retrieving room type list [deprecated]

This function is used to retrieve a list of all room types and codes in the specified language. This function should not be used anymore. Use the function DownloadExport instead. That function will give you all room types from SunHotels in one request.

Example URL with input:

<http://xml.sunhotels.net/14/PostGet/NonStaticXMLAPI.asmx/GetRoomTypes?userName=testagent&password=785623&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/14/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>
      <!--Indicates whether the room type is shared.-->
      <!--0 = Not shared, 1 = Shared -->
      <sharedRoom></sharedRoom>
      <!--Indicates whether facilities are shared for
      the room type (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>
```

Booking

This function is used to book a room/apartment/villa.

NOTE!

2 new endpoints have been added including search and book function and should be used from now. These new search and book functions include 2 new parameters, *customerCountry* and *B2C*, that will allow to dynamically advise client nationality and point of sale in search and booking request to SunHotels.

[See correct URLs to use for Search with Market and B2C parameters](#)

Example URL with input:

<http://xml.sunhotels.net/14/PostGet/NonStaticMarketSearch.aspx/Book?userName=testagent&password=785623¤cy=NOK&language=en&email=noreply@sunhotels.net&checkInDate=2011-04-23&checkOutDate=2011-04-25&roomId=2998&rooms=1&adults=1&children=1&infants=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=&customerCountry=gb&B2C=>

Function inputs

*: Required input

** Required input for adults and children included in booking

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.	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>*infants</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>yourRef</i>	A booking number or reference.	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).	Strings representing first and last names, e.g. "John" and "Smith"
<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 <= 11
<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 <= 11
...		

**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 $x \leq 11$
**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 $x \leq 11$
<i>customerEmail</i>	Email address of the customer. Non-required field. Only to be used in separate agreement with SunHotels.	String representing an email address.
<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.	"true"/"1" or "false"/"0".

Function outputs

Specification of output of function can be found at:

<http://xml.sunhotels.net/14/PostGet/NonStaticMarketSearch.aspx?op=Book>

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 meal that was booked (translated to the specified language)-->
    <board></board>
    <!--The meal that was booked (in English).-->
    <englishBoard></englishBoard>
    <!--The check-in date for the booking-->
    <checkindate></checkindate>
    <!--The check-out date for the booking-->
    <checkoutdate></checkoutdate>
    <!--The price of the booking-->
    <price></price>
    <!--The currency of the booked price-->
    <currency></currency>
    <!--The price of the booking (in Euros).-->
    <price.EUR></price.EUR>
    <!--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></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>
  </booking>
</bookResult>
```

```

<!--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.-->
<voucherURL></voucherURL>
<!--Text for the voucher associated with the booking-->
<voucherText></voucherText>
<!--An integer indicating whether a transfer has been booked
(1: yes, 0: no, 2: transfer has been cancelled)-->
<transferbooked></transferbooked>
<!--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).
NOTE:Important to inform client through booking process.-->
<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).
NOTE:Important to inform client through booking process.-->
<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).
NOTE:Important to inform client through booking process.-->
<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>

```

```

</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 1 to 11 years.	Modify the provided value so that it is a valid integer between 1 and 11.
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 1 and 11.
-----------------------------	--	--------------------------------

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/14/PostGet/NonStaticXMLAPI.aspx/CancelBooking?userName=testagent&password=785623&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/14/PostGet/NonStaticXMLAPI.asmx?op=CancelBooking>

XML output

```
<?xml version="1.0" encoding="utf-8"?>
<result>
  <!--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 booking.-->
  <Currency></Currency>
  <!-- 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>
  <!--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/14/PostGet/NonStaticXMLAPI.aspx/GetTransferTypes?userName=testagent&password=785623&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/14/PostGet/NonStaticXMLAPI.aspx?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/14/PostGet/NonStaticXMLAPI.aspx/GetThemes?userName=testagent&password=785623>

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/14/PostGet/NonStaticXMLAPI.aspx?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 [deprecated]

This function is used to retrieve a list of all destinations or destinations with a specific destination code. This function should not be used anymore. Use the function DownloadExport instead. That function will give you all destinations from SunHotels in one request.

Example URL with input:

<http://xml.sunhotels.net/14/PostGet/NonStaticXMLAPI.asmx/GetDestinations?userName=testagent&password=785623&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/14/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 [deprecated]

This function is used to retrieve a list of all resorts or resorts with a specific destination code or ID. This function should not be used anymore. Use the function DownloadExport instead. That function will give you all resorts from SunHotels in one request.

Example URL with input:

<http://xml.sunhotels.net/14/PostGet/NonStaticXMLAPI.asmx/GetResorts?userName=testagent&password=785623&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/14/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/14/PostGet/NonStaticXMLAPI.aspx/SearchTransfers?userName=testagent&password=785623&language=en¤cy=GBP&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>currency</i>	The currency that all prices will be converted to in the search results.	String representing a currency code, e.g. "GBP".
** <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

Function output

Specification of output of function can be found at:

<http://xml.sunhotels.net/14/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 price (in Euros)-->
              <priceEUR></priceEUR>
              <!--The rate price (in the specified currency)-->
              <price></price>
            </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>
  </Error>
</searchTransfersResult>
```

```
<!--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/14/PostGet/NonStaticXMLAPI.aspx/GetStaticHotelsAndRooms?userName=te
stagent&password=785623&language=en&destination=&hotelIDs=6928&resortIDs=&accommodation
Types=&sortBy=&sortOrder=&exactDestinationMatch=](http://xml.sunhotels.net/14/PostGet/NonStaticXMLAPI.aspx/GetStaticHotelsAndRooms?userName=te
stagent&password=785623&language=en&destination=&hotelIDs=6928&resortIDs=&accommodation
Types=&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/14/PostGet/NonStaticXMLAPI.asmx?op=GetStaticHotelsAndRooms>

XML Output

```
<?xml version="1.0" encoding="utf-8"?>
<getStaticHotelsAndRoomsResult>
  <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 beds that are available for the room-->
              <beds></beds>
              <!--The number of extra beds 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 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-->
        <sharedRoom></sharedRoom>
        <!--Whether facilities are shared for this room type-->
        <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 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 "http://www.sunhotels.net/SunHotels.net/Hote
lInfo/hotelImage.aspx ?id=239846&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 "http://www.sunhotels.net/SunHotels.net/Hote
lInfo/hotelImage.aspx ?id=239846")
    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>

```

```
        </distances>
      </distanceType>
    </distanceTypes>
    <!--The time zone where the accommodation is located-->
    <timeZone></timeZone>
    <!--Whether accommodation is "Best Buy" or not.-->
    <isBestBuy></isBestBuy>
  </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/14/PostGet/NonStaticXMLAPI.aspx/GetBookingInformation?userName=testagent&password=785623&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/14/PostGet/NonStaticXMLAPI.asmx?op=GetBookingInformation>

XML Output

```
<?xml version="1.0" encoding="utf-8" ?>
<getBookingInformationResult>
  <!--All retrieved bookings-->
  <bookings>
    <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 meal that was booked (translated to the specified language)-->
      <board></board>
      <!--The meal that was booked (in English).-->
      <englishBoard></englishBoard>
      <!--The check-in date for the booking-->
      <checkindate></checkindate>
      <!--The check-out date for the booking-->
      <checkoutdate></checkoutdate>
      <!--The price of the booking-->
      <price></price>
      <!--The currency of the booked price-->
      <currency></currency>
      <!--The price of the booking (in Euros).-->
      <price.EUR></price.EUR>
      <!--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></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.-->
  <voucherURL></voucherURL>
  <!--Text for the voucher associated with the booking-->
  <voucherText></voucherText>
  <!--An integer indicating whether a transfer has been booked
(1: yes, 0: no, 2: transfer has been cancelled)-->
  <transferbooked></transferbooked>
  <!--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).
NOTE:Important to inform client through booking process.-->
  <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).
NOTE:Important to inform client through booking process.-->
  <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).
NOTE:Important to inform client through booking process.-->
  <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>
  </englishRoomNotes>

```

```
        </englishRoomNote>
    </englishRoomNotes>
    <!--The status of the booking-->
    <bookingstatus>NotActive or Active</bookingstatus>
</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/14/PostGet/NonStaticXMLAPI.asmx/GetTransferBookingInformation?userName=testagent&password=785623&language=en&bookingID=2437648&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/14/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>
    </transferBooking>
  </transferBookings>
</getTransferBookingInformationResult>
```

```
<!--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>
</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/14/PostGet/NonStaticXMLAPI.aspx/AddTransfer?userName=testagent&password=785623&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>

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>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 = Arrival transfer only 1 = Arrival and departure transfer 2 = Departure transfer only
<i>*returnAirline</i>	The name of the airline used for the return transfer Note: This is only required if " <i>returnTransfer</i> " 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 " <i>returnTransfer</i> " is set to 1	String representing a flight number
<i>*returnDepartureDate</i>	Date of departure Note: This is only required if " <i>returnTransfer</i> " 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 " <i>returnTransfer</i> " is set to 1	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/14/PostGet/NonStaticXMLAPI.asmx?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>
</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.

Canceling a transfer booking

This function is used to cancel an existing transfer booking.

Example URL with input:

<http://xml.sunhotels.net/14/PostGet/NonStaticXMLAPI.aspx/CancelTransferBooking?userName=testagent&password=785623&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/14/PostGet/NonStaticXMLAPI.asmx?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/14/DownloadExport.aspx?username=testagent&password=785623&language=en&xmlType=Place&version=140>

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.	A valid ISO code from the "GetLanguages" function.
<i>version</i>	The version of the XML content. 140 have to be specified to get the most current resort mapping. The value 140 stands for the schema 1.4.0. Newer versions will be released in the future.	140

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_1.4.0.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_1.4.0.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>environment</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>
    </type>
  </room_types>
</sunhotels_cache_file>
```

```

    <shared_facilities>>false</shared_facilities>
  </type>
</room_types>
<cancellation_policies />
<hotels>
  <hotel>
    <id>535</id>
    <name>Niu Daus</name>
    <accomodation_type>apartment</accomodation_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>
<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="7472">
    <image_variant>http://www.sunhotels.net/SunHotels.net/HotelInfo/hotelImage.aspx?id=7472</image_variant>
  </image>
  <image id="7471">
    <image_variant>http://www.sunhotels.net/SunHotels.net/HotelInfo/hotelImage.aspx?id=7471</image_variant>
  </image>
  <image id="35725">
    <image_variant>http://www.sunhotels.net/SunHotels.net/HotelInfo/hotelImage.aspx?id=35725</image_variant>
  </image>
  <image id="35724">
    <image_variant>http://www.sunhotels.net/SunHotels.net/HotelInfo/hotelImage.aspx?id=35724</image_variant>
  </image>
  <image id="35723">
    <image_variant>http://www.sunhotels.net/SunHotels.net/HotelInfo/hotelImage.aspx?id=35723</image_variant>
  </image>
  <image id="7470">
    <image_variant>http://www.sunhotels.net/SunHotels.net/HotelInfo/hotelImage.aspx?id=7470</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>
    <best_buy>true</best_buy>
  </room>
  <room type_id="52.4.0">
    <room_id>15972</room_id>
    <best_buy>>false</best_buy>
  </room>
</rooms>
</hotel>
</hotels>
</sunhotels_cache_file>

```

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.

As of the writing of this document, the following languages are supported:

English, Spanish, Swedish, German, French, Finnish, Polish, Norwegian and Danish.

Example URL with input:

<http://xml.sunhotels.net/14/PostGet/StaticXMLAPI.asmx/GetLanguages?userName=testagent&password=785623>

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/14/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/14/PostGet/NonStaticXMLAPI.asmx/GetMeals?userName=testagent&password=785623&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/14/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/14/PostGet/NonStaticXMLAPI.asmx/GetFeatures?userName=testagent&password=785623&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/14/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>
```