



## **Viator API Documentation - Partner**

Version 1.24

© 2015 Viator Inc

# Table of Contents

<b>Part I Viator API Overview</b>	<b>4</b>
<b>Part II Caching API Results</b>	<b>5</b>
<b>Part III Connectivity</b>	<b>5</b>
<b>Part IV Common Workflows and Data Validation</b>	<b>5</b>
1 Add to Cart.....	5
2 Checkout.....	8
3 View Voucher or Confirmation Status.....	10
<b>Part V APIs</b>	<b>13</b>
1 Utility APIs.....	13
/util/weather .....	13
/util/ip2country .....	15
/util/version .....	15
/util/sitedetails .....	15
/util/countrymap .....	16
2 Taxonomy APIs - Destination and Categories.....	19
/taxonomy/locations .....	19
/taxonomy/categories .....	21
3 Product APIs.....	24
/product .....	24
/product/reviews .....	38
/product/photos .....	41
4 Search APIs.....	43
/search/products .....	43
/search/products/codes .....	49
/search/freetext .....	51
5 Booking APIs.....	56
Booking Data Classes .....	58
/booking/availability/dates .....	59
/booking/availability .....	61
/booking/availability/tourgrades .....	62
/booking/availability/tourgrades/pricingmatrix .....	66
/booking/calculateprice .....	70
/booking/hotels .....	74
/booking/book - partners .....	77
Test Bookings for API partners.....	88
Errors .....	89
/booking/status .....	90
/booking/pricingmatrix .....	93
/booking/voucher .....	97
/booking/voucher/email .....	99
/booking/voucher/fax .....	100
/booking/pastbooking .....	101
/booking/mybookings .....	103
/booking/mybookings/message .....	106

6	Support APIs.....	108
	/support/terms .....	108
	/support/faq .....	108
	/service/support/customercare .....	109
<b>Part VI Sample Code</b>		<b>109</b>
1	PHP Sample Code.....	109
	POST Request .....	109
	GET Request .....	110
2	Java Sample Code.....	111
	Java POST Request .....	111
<b>Part VII Appendix</b>		<b>111</b>
1	Standard JSON fields.....	112
2	Country Codes.....	113
3	USA States.....	117
4	Canada Provinces.....	118
5	Australian States.....	118
6	Booking and Item Status List.....	118
7	Required Info Questions.....	119
8	Supported currency codes.....	120
<b>Index</b>		<b>0</b>

# 1 Viator API Overview

The Viator API is designed to provide internal Viator projects and partners with a set of API calls to enable the construction of a complete tours and activities website, application or similar for integration into existing systems or the creation of new channels.

The complete list of Viator products, destinations, user submitted photos & reviews, pricing and availability are all available through the API.

The API is divided into several sections - Utility, Search, Destination, Product, Booking, Attraction/Recommendation, Storage, User content submission, and User Account APIs.

Viator Partners can use all but the Storage, User Content Submission and User Account APIs as it is assumed that they have their own systems for this functionality. The documentation for these APIs are not provided to the partner. Access to Attraction/Recommendation is on a per contract basis.

The API Service is delivered via HTTP(s) POST and GET requests. The POST request data and the returned data is in JSON (Javascript Object Notation) format. This allows for very simple serialization/deserialization with most modern languages.

## Service Directory

There is a services directory at <http://prelive.viatorapi.viator.com/service/directory>

The directory can be browsed and used to experiment with live data. If you use FireFox and install JSONView you get formatted output making exploration simple.

Be sure to enter the API key which is saved for the session.

**Viator API 1.5.9 service directory**  
API Key:

**API Key Details**  

ID:	123456711
Company Name:	JJ Travel
Retail Site Id:	1234.api.partner.viator.com
Default Currency:	USD
Locale:	en

[IMPORTANT: Please note \(click here to expand/collapse\)](#)  
[Example codes \(click here to expand/collapse\)](#)

Service	Description	Incoming JSON/Params (double click to expand/collapse)
<a href="#">/service/util/version</a>	GET	Shows the version Info to see if we are up
<a href="#">/service/util/sitedetails</a>	GET	Show the site details for the APIKey provided
<a href="#">/service/util/ip2country</a>	GET	Determines the country from the ip address. If param passed then uses that else the ip from request  ip=xxx
<a href="#">/service/util/countrymap</a>	GET	Get a country map used with Credit cards etc. A map of the country code to country details
<a href="#">/service/util/weather</a>	GET	Gets the weather given destination id and optionally the temperature unit  destId=6846tempUnit=C

```

{
  - data: [
    - {
      duration: "3 hours 30 minutes",
      shortDescription: "Take off from Las Vegas Airport on an exhilarating helicopter flight to the Grand Canyon. You'll enjoy a 45-minute helicopter flight each way, land deep in the canyon for a champagne picnic, and fly low over the famous Las Vegas neon Strip on your return. The Grand Canyon All American Helicopter Tour is a half-day tour, leaving you with time in the day free to explore Las Vegas and Nevada at your leisure.",
      currencyCode: "USD",
      supplierCode: "2280",
      code: "2280AAHT",
      title: "Grand Canyon All American Helicopter Tour",
      - catIds: [
        1,
        5
      ],
      thumbnailURL: http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs154x109/2280/SITours/grand-canyon-all-american-helicopter-tour-in-las-vegas-115053.jpg,
      supplierName: null,
      photoCount: 765,
      reviewCount: 2812,
      specialReservation: false,
      rating: 5,
    }
  ]
}

```



### API Key

You will require an API Key in order to use the API. Please request this from the Viator Business Development team via our partner site at <http://partner.viator.com>.

The API key must be appended to every API call as a URL parameter for both GET and POST calls.

The API key is associated with a currency, language and partner IDs for Viator Partner purposes.

The best way to get familiar with the API is to explore the service directory.

## 2 Caching API Results

All users of the API will be monitored for request traffic.

Calls to largely static data such as Taxonomy, Product Lists, Product Details and Attractions/Recommendations should be cached as it will improve your customers' experience and reduce the traffic and load on Viator's links and servers. Cache expiry times are recommended to be 6 to 12 hours - change to 24 hours

Booking APIs and Freetext search APIs should not need to be cached as the data is dynamic and regularly updated with changes to availability, pricing, etc.

Viator utilizes memcache for our web application service consumers and we use local file or database storage for mobile applications.

We have a separate document on caching strategy - please request this from the BD team if you would like to read it.

## 3 Connectivity

### Latency

30 second timeout on both PreLive and Live

### Rate limiting

PreLive - 15 requests in 10 seconds. Requests that exceed this will get a 429 error (Too Many Requests)

Live - 45 requests in 10 seconds. Requests that exceed this limit will be moved to a different pool of servers and may be dropped when the pool is full.

## 4 Common Workflows and Data Validation

The workflows commonly adopted by users of the API tend to be around the booking process. The common workflows are Add to Cart, Checkout and View Voucher or Confirmation Status.

### 4.1 Add to Cart

#### Summary

The "cart" must be managed by the API consumer as the API is stateless and has no concept of a cart. It can generate a price quote for one or several items, but this collection will not be saved at the

server.

The recommended process is as follows

1. View Product - the consumer is viewing product details
2. Select date and Passengers - they then check available dates, select passenger types and qty.
3. Select Tourgrade - available tourgrades are listed
4. View Cart, Price Quote - they get a final price (including multiple items taking into account any promotion codes)

## View Product

From the view product details, view product schedule, etc a button is presented to "check availability" or "book now" or similar. Use the [/product](#) API for product details and schedule.

## Select Date and Passengers

Here the customer is presented with the dates that there is any availability using the [/booking/availability/dates](#) API. The list of age bands (age band names ie: Adult, Child and each band's min/max age) is retrieved from the [/product](#) API.

### Validation

1. no more than 9 passengers total
2. at least one adult (or adult equivalent) passenger selected.
3. check the date to be selected from the [/booking/availability/dates](#) API so it should have availability.

## Select Tourgrade

Once the date and passenger mix have been selected, the consumer may need to select a tourgrade. Products that do not have a tourgrade in the Viator system are given a tourgrade of DEFAULT in the API. For the products that only have one tourgrade of DEFAULT, this can simply be assumed and this step can be skipped.

For products with multiple tourgrades (ie: different departure times, passenger mix price deals ie: family pass, different inclusions and price ie: limo pickup, different language delivery options), or non-default tourgrades, these tour grades must be presented to the consumer.

The tourgrades are retrieved with the [/booking/availability/tourgrades](#) API call. If the tourgrade is available (check the available flag), an add to cart button can be included. For unavailable tourgrades an unavailableReason is provided of TRAVELLER\_MISMATCH, BLOCKED\_OUT or UNAVAILABLE. You may choose to hide the blocked out grades (if a day has all blocked out grades or is not operating that day would not have been returned from the [/booking/availability/dates](#) API anyways). You may get a result that has no bookable tour grades on a day if the passenger mix does not meet that tourgrade passenger mix restrictions. For example, a wedding might require 2 adults as the only possible passenger mix.

If a tour grade is unavailable (available != 1 ) it will be for one of these 3 reasons (unavailableReason):

TRAVELLER\_MISMATCH  
BLOCKED\_OUT  
UNAVAILABLE

You can choose to hide the BLOCKED\_OUT tour grades but you will probably want to display the TRAVELLER\_MISMATCH grades with the passenger mix requirements listed so the consumer can elect to alter their passenger mix to suit (ie: for family passes, etc) You cannot allow the passenger

to book with an incompatible passenger mix.

The return object includes:  
ageBandsRequired

- \* MinimumCountRequired - Minimum count of travellers for an ageband
- \* MaximumCountRequired - Maximum count of travellers for an ageband. If null then any

You will need to present the ageBandsRequired information as in the following examples:

if in Adult age band:

```
MinimumCountRequired = 1  
MaximumCountRequired = 3  
from one to 3 adults
```

if in Adult age band:

```
MinimumCountRequired = 0  
MaximumCountRequired = 3  
up to 3 adults
```

if in Adult age band:

```
MinimumCountRequired = 3  
MaximumCountRequired =  
3 or more adults
```

if in Adult age band:

```
MinimumCountRequired = 2  
MaximumCountRequired = 2  
2 adults
```

Note that there is a singular and plural version for each age band so you can add the if statements to make the samples above grammatically correct.

The user can press back and change their passenger mix or try selecting another day.

## View Cart / Price Quote

Once the item has been added to the cart you will need to save this data in the session or a cookie. The data that will need to be stored is the product code, and age band to quantity passenger mix.

Sample PHP shopping cart item class (note this is an example and is implementation dependant - you could use the item class (as used in the API calls) for this but it would take up extra memory if stored in a cookie, etc):

```
class shoppingCartItem  
{  
    var $code;  
    var $tourGrade;  
    var $date;  
    var $ageBandIdToQty;          // assoc array  
}
```

To get a price quote, you must call the [/booking/calculateprice](#) API. This API requires a currency code, optional promo code and an array of items. Each item contains the date, product code, tour grade code and an associative array of ageBand IDs to quantity.

### Validation



The data must contain valid agebands, tour codes, dates, etc as returned from the API calls made

previously. Note that availability can change between API calls as tour grades or products may be blocked out or the booking window may close for upcoming dates.

## 4.2 Checkout

### Summary

The checkout process has only one main API call - [/booking/book](#) but you may need to make calls to calculate prices, display product information (ageband names, etc) and list available pickup hotels for user selection.

The Viator mobile web site breaks the workflow down into 3 steps. For multiple items, some steps would need to be repeated. Your implementation can reorganize/reorder the data collection to better meet your needs.

#### Example workflow:

Traveler Details - collect the names of the lead traveler and all other travelers

Travel Details - ask additional questions, special requirements and select pick up options

(Note the steps above need to be iterated for each item)

Booking Details - get billing names, address, credit card details and accept terms and conditions.

The data collection details and validation will be organised into the steps above, but as mentioned, you can alter the workflow to meet your needs.

The classes for the booking request are defined here in [Booking Data Classes](#). You will need implement these in your chosen programming language and validate the correct JSON objects are generated during serialization.

### Traveler Details

The traveler details are used to populate the booking->items[]->travellers[] objects.

One passenger must be identified as the "Lead Traveler". There is a boolean on the traveler object for this flag. The lead traveler must be an Adult or have the ageband treatAsAdult flag set to true.

All other travelers must be included.

#### Validation

- bandId - must be a valid age band ID for the product
- firstname - must be 1-15 chars
- surname - must be 2-35 chars
- title - optional
- leadTraveller - boolean and one of the travelers must be true. That traveler must be in an ageband that has the treatAsAdult flag set to true.
- cellPhoneCountryCode - optional
- cellPhone - optional but recommended. Length must be 0 or 6-25 chars.

### Travel Details

#### Booking Questions





The travel details include the booking questions which are supplied in the [/product](#) API.

### Sample question

Note: There may be more than one.

```
bookingQuestions: [
  {
    message: "For safety reasons you must enter the weight of <b>all</b> passe
    required: true
    questionId: 100
    title: "Passenger Weights"
    subTitle: "(e.g. 127 pounds, 145 kilos, etc)"
    sortOrder: 1
  }
]
```

The questions should be displayed with the title, message, subtitle and whether it is mandatory (required) or not.

### Validation

If the question is mandatory, the user must enter at least one character.

### Special Requirements

Special Requirements also need to be presented as a text box. Customers may enter that they require wheelchair assistance, etc. It is not mandatory for the customer to enter any text.

### Pickup Information

The last thing that must be collected for each item being booked is the pickup information. If the product includes pickup the `hotelPickup` flag will be set to true (on the product object).

If pickup is included you will need to call the [/booking/hotels](#) API to determine if a hotel list exists. If yes, it must be displayed for selection. If not, a text box is displayed for hotel name collection.

Please note that the first three results in the list are not hotels but are alternate selections comprising:

- Hotel not listed
- Live locally or staying at family/friends
- Hotel not booked yet

If the user selects Hotel not listed, you must provide a text box for them to enter a hotel. For the other two options, no hotel name is required. In all cases the hotel ID must be updated with either a hotel ID or the IDs of the three items listed.

If no hotel list is available, you must provide a text box for collecting their hotel name. Please include instructions to enter *Live locally* or *Hotel not booked yet* if they cannot provide a hotel name.

### Validation

If hotel list is available, the `hotelID` must be supplied and if the `hotelID` is 'notListed' the `pickupPoint` field must have at least one character.

If a hotel list is not available, then `pickupPoint` must have a value.

### Billing Information



The billing or booking information step is where the credit card details are collected.

### Validation

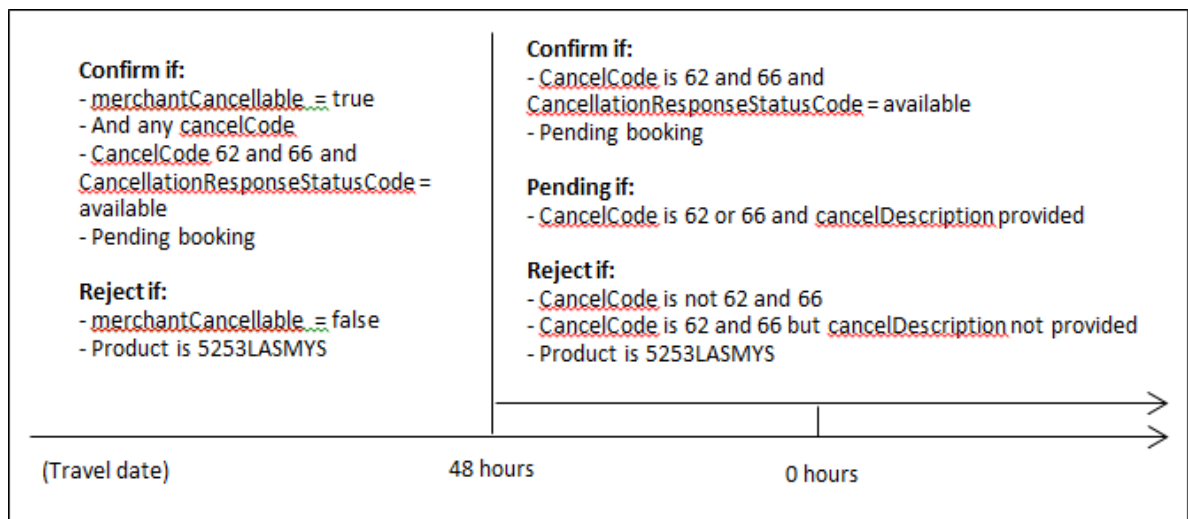
- ccaddress1 - at least 1 char
- ccaddress2 - none
- ccaddressCity - at least one char
- ccaddressState - at least one char. from list in appendix if country is Canada, USA or Australia.
- ccaddressZip - at least one char
- ccphone - none
- cctype - one of Visa, Mastercard or Amex
- ccname - this must be at least two words (first and last name) and 2 chars long
- ccnumber - this must be valid. you must use a luhn check to validate
- ccaddressCountryId - must be a country code from the list in appendix
- ccexpire -
- ccadditionalDigits - must be 3 char numbers for Visa and Mastercard or 4 for American Express

### Accept Viator Terms and Conditions

This must be a checkbox and it **must** be checked prior to booking API submission. Place it at the bottom of the checkout page, before the button to submit the payment.

Please note that the value of the checkbox is not sent to Viator. It is only for front end validation.

Example:



## 4.3 View Voucher or Confirmation Status

The result object returned from service /booking/book holds booking details that can be used for displaying an order summary to the user.

Confirmed bookings (freesale products) will return a voucherKey / voucherURL at an itinerary and item level which can be accessed by the customer to view their voucher.

Pending bookings (on request products) will return a null voucherKey / voucherURL, and will only contain a value once the booking is confirmed.



A bookingStatus object is included in the result object which contains the status of the booking. It is available on an itinerary level and an item level.

```
bookingStatus: {
  type: "CONFIRMED",
  level: "ITINERARY",
  failed: false,
  text: "Confirmed",
  cancelled: false,
  status: 3,
  confirmed: true,
  amended: false,
  pending: false
}
```

The status field corresponds to a number in the [Booking and Item Status List](#) for the level.

Flags - failed, cancelled, confirmed, amended, pending - can also be checked for the status of the item.

Pending bookings - a booking is pending if it is in progress. For example, an On Request booking would be pending until it is confirmed/rejected by the supplier. A booking would be pending amendment if a customer made an amendment to an on request booking that is yet to be accepted by the supplier. The booking would then have a status of amended when the supplier or customer service accepts the amendment.

#### Example result object

```
{
  "errorReference": null,
  "data": {
    "sortOrder": 0,
    "rulesApplied": [],
    "bookingStatus": {
      "status": 3,
      "text": "Confirmed",
      "type": "CONFIRMED",
      "level": "ITINERARY",
      "confirmed": true,
      "pending": false,
      "amended": false,
      "cancelled": false,
      "failed": false
    },
    "itemSummaries": [{
      "sortOrder": 0,
      "rulesApplied": [],
      "bookingStatus": {
        "status": 1,
        "text": "Paid & Confirmed",
        "type": "CONFIRMED",
        "level": "ITEM",
        "pending": false,
        "failed": false,
        "confirmed": true,
        "amended": false,
        "cancelled": false
      },
      "travellerAgeBands": [{
        "sortOrder": 0,
        "count": 1,
        "pluralDescription": "Adults",
        "description": "Adult",

```

```

        "ageBandId": 1
    }],
    "voucherKey": "1000024692:db87465199ec5c70f9eb56dd5ad0482e6eb86298f9a7c51283a223c",
    "voucherURL": "https://prelive.partner.viator.com/voucher.jspa?PUID=1010&setLocal",
    "voucherRequirements": "You must present a paper voucher for this tour. We",
    "productPulledDown": false,
    "merchantCancellable": false,
    "productWidgetList": null,
    "savingAmount": 0,
    "passbooks": null,
    "termsAndConditions": null,
    "itineraryId": 1000024692,
    "tourGradeCode": "1DAYBOAT",
    "productCode": "2065CPT",
    "leadTravellerSurname": "Test",
    "distributorItemRef": null,
    "languageServicesLanguageCode": "en",
    "travelDate": "2016-02-01",
    "price": 4.51,
    "bookingEngineId": "UF",
    "merchantNetPrice": 0,
    "merchantNetPriceFormatted": "",
    "leadTravellerTitle": "Mr",
    "leadTravellerFirstname": "Homer",
    "lastRetailPrice": 4.51,
    "destId": 318,
    "voucherOption": "VOUCHER_PAPER_ONLY",
    "productTitle": "Cape Town City Hop-on Hop-off Tour",
    "itemId": 700025504,
    "barcodeOption": "perperson",
    "barcodeType": "code128",
    "obfsId": 9779,
    "priceFormatted": "$4.51",
    "savingAmountFormatted": "$0.00",
    "priceUSD": 4.51,
    "lastRetailPriceFormatted": "$4.51",
    "departsFrom": "Cape Town, South Africa",
    "hoursConfirmed": 0,
    "currencyCode": "USD"
}],
    "voucherURL": "https://prelive.partner.viator.com/voucher.jspa?PUID=1010&setLocal",
    "voucherKey": "1000024692:db87465199ec5c70f9eb56dd5ad0482e6eb86298f9a7c51283a223c",
    "bookerEmail": "apitest@viator.com",
    "userId": null,
    "itineraryId": 1000024692,
    "exchangeRate": 1,
    "totalPriceFormatted": "$4.51",
    "totalPriceUSD": 4.51,
    "bookingDate": "2015-03-17",
    "distributorRef": null,
    "totalPrice": 4.51,
    "hasVoucher": true,
    "currencyCode": "USD"
},
    "dateStamp": "2015-03-17T17:42:21+0000",
    "errorType": null,
    "errorMessage": null,
    "errorName": null,
    "success": true,
    "totalCount": 1,
    "vmid": "321003",
    "errorMessageText": null
}

```

Checking the status of one booking:

Service `/booking/mybookings` can be used to check the status of a booking post-purchase. This is useful for checking the status of a pending booking, particularly if there are multiple items within the booking.

It is recommended that you poll the service not more than once an hour.

Checking the status of multiple bookings:

Service `/booking/status` will return the status of all bookings based on a booking date range, travel date range, specific distributor/distributor item references, and lead traveller date/first name. This is useful for checking the status of pending bookings that have been made recently or that will be traveled on soon.

You can only poll this service (successful calls) once every half an hour.

## 5 APIs

The APIs for making a booking, searching for a product, retrieving the location and category taxonomy etc. are listed in detail in this section.

### 5.1 Utility APIs

Utility APIs include API version, weather, IP to Country, site details and country code to country details mapping.

#### 5.1.1 `/util/weather`

Method: POST

This is a simple API to get the weather for a Viator destination.

Sample URL parameters:  
`destId=684&tempUnit=C`

##### **tempUnit**

Either:

- C - Celcius
- F - Fahrenheit

##### **List of weather descriptors**

SUNNY  
CLEAR  
MOSTLY SUNNY  
MOSTLY CLEAR  
HAZY SUNSHINE  
HAZE  
PASSING CLOUDS  
MORE SUN THAN CLOUDS  
SCATTERED CLOUDS  
PARTLY CLOUDY  
A MIXTURE OF SUN AND CLOUDS  
HIGH LEVEL CLOUDS



MORE CLOUDS THAN SUN  
PARTLY SUNNY  
BROKEN CLOUDS  
MOSTLY CLOUDY  
CLOUDY  
OVERCAST  
LOW CLOUDS  
LIGHT FOG  
FOG  
DENSE FOG  
ICE FOG  
SANDSTORM  
DUSTSTORM  
INCREASING CLOUDINESS  
DECREASING CLOUDINESS  
CLEARING SKIES  
BREAKS OF SUN LATE  
EARLY FOG FOLLOWED BY SUNNY SKIES  
AFTERNOON CLOUDS  
MORNING CLOUDS  
SMOKE  
LOW LEVEL HAZE

#### Example Result Object

```
{
  "data": {
    "metric": true,
    "dew_point": 18,
    "baro_tendency": "rising",
    "currentTime": "5:31 PM Monday, Sep 3",
    "temperature": "11&#176 C",
    "wind_dir": "NWN",
    "humidity": 88,
    "dew_Point": "18.0&#176 C",
    "comfort": "20.0&#176 C",
    "baro_pressure": "1018.2mb",
    "visibility": "16.09km",
    "description": "Sunny. Cool.",
    "icon": "1.gif",
    "daylight": false,
    "temp_desc": "temp desc",
    "localupdatetime": "Mon, 10 Dec 2007 13:57:55 PST",
    "sky_desc": "SUNNY",
    "precip_desc": "precip desc",
    "air_desc": "Air desc",
    "wind_speed": "9.26 kph"
  },
  "vmid": "321002",
  "errorMessage": null,
  "errorType": null,
  "dateStamp": "2012-09-03T17:31:40+0000",
  "errorReference": null,
  "errorMessageText": null,
  "success": true,
  "totalCount": 1,
  "errorName": null
}
```



### 5.1.2 /util/ip2country

Method: POST

This API returns a country based on the user's IP address.

Get parameters

ip=xxx.xxx.xxx.xxx

#### Example Result Object

```
{
  data: "AU",
  vmid: "321002",
  errorMessage: null,
  errorType: null,
  dateStamp: "2012-09-03T17:33:40+0000",
  errorReference: null,
  errorMessageText: null,
  success: true,
  totalCount: 1,
  errorName: null
}
```

### 5.1.3 /util/version

Method: POST

Get the version of the current API build.

No parameters

#### Example Result Object

```
{
  data: {
    currentTime: "Mon Sep 03 17:37:16 PDT 2012",
    serverStartTime: "Tue Aug 28 22:28:59 PDT 2012",
    vmid: "321002",
    serviceVersion: "1.5.0.99-20120823",
    buildTimestamp: "2012-08-23 14:08:58",
    vTravelVersion: "Build Build188.189-20120823 (2012-08-23 12:19:39)",
    upTime: "5 days, 139 hours, 8348 minutes, 500897 seconds"
  },
  vmid: "321002",
  errorMessage: null,
  errorType: null,
  dateStamp: "2012-09-03T17:37:16+0000",
  errorReference: null,
  errorMessageText: null,
  success: true,
  totalCount: 1,
  errorName: null
}
```

### 5.1.4 /util/sitedetails

Method: GET

Get the site details based on the API key passed



No parameters

### Example Result Object

```
{
  data: {
    url: "prelive.partner.viator.com",
    id: 10388573996242374,
    channel: "PARTNER",
    enabled: true,
    locale: "en",
    puid: "5413",
    retailSiteId: "5413.api.partner.viator.com",
    companyName: "Travelsmoove",
    defaultCurrency: "USD",
    centralURL: "prelive.partner.viator.com",
    permissionsDenied: [
      53,
      22,
      42,
      26,
      46,
      15
    ]
  },
  vmid: "321002",
  errorMessage: null,
  errorType: null,
  dateStamp: "2012-09-03T17:39:01+0000",
  errorReference: null,
  errorMessageText: null,
  success: true,
  totalCount: 1,
  errorName: null
}
```

## 5.1.5 /util/countrymap

Method: GET

This API returns a list of countries mapped to country and phone codes. This is useful when validating credit cards.

No parameters

### Example Result Object

```
{
  data: {
    AF: {
      countryName: "Afghanistan",
      phoneCode: "",
      countryId: "AF"
    },
    AU: {
      countryName: "Australia",
      phoneCode: "+61",
      countryId: "AU"
    },
    AT: {
      countryName: "Austria",
      phoneCode: "+43",
      countryId: "AT"
    }
  }
}
```





```
AZ: {
  countryName: "Azerbaijan",
  phoneCode: "",
  countryId: "AZ"
},
BS: {
  countryName: "Bahamas",
  phoneCode: "",
  countryId: "BS"
},
BH: {
  countryName: "Bahrain",
  phoneCode: "+973",
  countryId: "BH"
},
BD: {
  countryName: "Bangladesh",
  phoneCode: "",
  countryId: "BD"
},
BB: {
  countryName: "Barbados",
  phoneCode: "",
  countryId: "BB"
},
BY: {
  countryName: "Belarus",
  phoneCode: "",
  countryId: "BY"
},
BE: {
  countryName: "Belgium",
  phoneCode: "+32",
  countryId: "BE"
},
BR: {
  countryName: "Brazil",
  phoneCode: "+55",
  countryId: "BR"
},
AE: {
  countryName: "United Arab Emirates",
  phoneCode: "+971",
  countryId: "AE"
},
GB: {
  countryName: "United Kingdom",
  phoneCode: "+44",
  countryId: "GB"
},
US: {
  countryName: "United States of America",
  phoneCode: "+1",
  countryId: "US"
},
UY: {
  countryName: "Uruguay",
  phoneCode: "",
  countryId: "UY"
},
UM: {
  countryName: "US Minor Outlying Islands",
  phoneCode: "",
  countryId: "UM"
},
UZ: {
```



```

        countryName: "Uzbekistan",
        phoneCode: "",
        countryId: "UZ"
    },
    VU: {
        countryName: "Vanuatu",
        phoneCode: "",
        countryId: "VU"
    },
    VE: {
        countryName: "Venezuela",
        phoneCode: "+58",
        countryId: "VE"
    },
    VN: {
        countryName: "Vietnam",
        phoneCode: "",
        countryId: "VN"
    },
    VG: {
        countryName: "Virgin Islands-British",
        phoneCode: "",
        countryId: "VG"
    },
    VI: {
        countryName: "Virgin Islands-US",
        phoneCode: "",
        countryId: "VI"
    },
    WF: {
        countryName: "Wallis and Futuna Islands",
        phoneCode: "",
        countryId: "WF"
    },
    WS: {
        countryName: "Western Samoa",
        phoneCode: "",
        countryId: "WS"
    },
    YE: {
        countryName: "Yemen Republic",
        phoneCode: "",
        countryId: "YE"
    },
    ZM: {
        countryName: "Zambia",
        phoneCode: "",
        countryId: "ZM"
    },
    ZW: {
        countryName: "Zimbabwe",
        phoneCode: "",
        countryId: "ZW"
    }
},
vmid: "221002",
errorMessage: null,
errorType: null,
dateStamp: "2012-04-11T17:30:18+0000",
success: true,
errorReference: null,
errorMessageText: null,
totalCount: 1,
errorName: null
}

```



## 5.2 Taxonomy APIs - Destination and Categories

Viator support a limited number of destinations. You may need to map our destination IDs to your destination taxonomy.

Our destination taxonomy contains parentIDs so you can create a map in memory for traversing these relationships and displaying appropriate content.

When a search is completed it includes the destination supplied as well as children destinations automatically.

### 5.2.1 /taxonomy/locations

Method: GET

This returns a complete list of Viator destinations including destination names and parent IDs. Common usage is to provide navigation through drill down lists or combo boxes.

#### URL Parameters

No parameters

The screenshot shows the top navigation bar of the Viator website with a search input field and a 'Go' button. Below this is a breadcrumb trail: 'Home > Select Destination'. The main content area is titled 'Select Destination' and includes instructions: 'Jump to any Viator destination by selecting your country and/or city and tapping 'Go''. There are two dropdown menus: the first is set to 'Australia' and the second is labeled 'Select a City or Region'. A 'Go' button is next to the second dropdown. At the bottom, there is a navigation bar with icons for Home, Search, Wishlist, Account, and Currency. The footer contains links for 'View Full Site', 'About', 'Help', 'Low-Price Guarantee', and 'Privacy', along with the Viator logo and copyright information: '© 1997–2011 Viator, Inc. Terms & Conditions'.

#### API Return Object

```
{  
  "errorReference": null,  
  "data": [  
    {  
      "sortOrder": 1,  

```



```

        "selectable": false,
        "defaultCurrencyCode": "USD",
        "iataCode": null,
        "timeZone": "America/New_York",
        "destinationType": "COUNTRY",
        "destinationName": "USA",
        "destinationId": 77,
        "parentId": 8,
        "lookupId": "8.77",
        "latitude": 37.09024,
        "longitude": -95.712891
    },
    {
        "sortOrder": 1,
        "selectable": true,
        "defaultCurrencyCode": "USD",
        "iataCode": null,
        "timeZone": "America/Anchorage",
        "destinationType": "REGION",
        "destinationName": "Alaska",
        "destinationId": 270,
        "parentId": 77,
        "lookupId": "8.77.270",
        "latitude": 63.588753,
        "longitude": -154.4930619
    },
    {
        "sortOrder": 2,
        "selectable": true,
        "defaultCurrencyCode": "USD",
        "iataCode": "ANC",
        "timeZone": "US/Alaska",
        "destinationType": "CITY",
        "destinationName": "Anchorage",
        "destinationId": 4152,
        "parentId": 270,
        "lookupId": "8.77.270.4152",
        "latitude": 61.2180556,
        "longitude": -149.9002778
    }
],
"dateStamp": "2014-11-24T21:30:47+0000",
"errorType": null,
"errorMessage": null,
"errorName": null,
"success": true,
"totalCount": 1,
"vmid": "321050",
"errorMessageText": null
}

```

Description of elements in JSON Return object



Element	Type	Comments	To be viewed by consumer	Required field?
data	object	Object that holds the destination data	N	Y
timeZone	varchar	The timezone of the destination	Optional	Y
destinationType	varchar	destination type: COUNTRY - Destination is a country REGION - Destination is a recognized region of a country, such as a state, Island within a country, stretch of coastline, etc CITY - Destination is a city	Optional	Y
destinationName	varchar	The name of the destination	Y	Y
destinationId	smallint	The Id of the destination. This Id is used throughout the services	N	Y
parentId	smallint	The Id of the parent destination. The parentId for a city can be either a region or a country. The parentId for a region would be a country. The parentId for a country would be a continent. Continents are not returned in this service.	N	Y
lookupId	varchar	consists of the parentId/s and destinationId. <top level parent Id>.<any additional parentIds>.<destinationId> e.g. 8.77.673 for Chicago	N	Y
latitude	decimal	Latitude coordinate of the destination	Optional	Y
longitude	decimal	Longitude coordinate of the destination	Optional	Y
selectable	boolean	An internal field that is no longer being used.	N	Y
defaultCurrencyCode	varchar	Default currency of the destination. Destinations with standard currencies will have their correct currency listed. For smaller countries, their default will be USD. e.g. Vietnam would be USD, Australia would be AUD.	Optional	Y
sortOrder	smallint	Suggested sort order of the destination within the parentId.	N	Y
iataCode	varchar	The IATA Code of the airport within the city. The IATA code is not available for destinations with a destination type of COUNTRY or REGION as there could be many within the destination.	N	Y

### 5.2.2 /taxonomy/categories

Method: GET

The category taxonomy in the Viator API consists of two levels - category and subcategories.

This service can be used to get a list of all categories and sub-categories. You can pass the destId to get a list of categories and sub-categories for a single destination, or omit the destId to get a list of all categories and sub-categories. The categories and subcategories are returned by sortOrder. These

category and subcategory IDs can be subsequently used for the search/products API call to filter products.

If a destId is not passed, a null productCount and thumbnailURL is returned as they are destination specific fields.

### URL Parameters

destId=684

### Description of URL parameters

Parameter	Type	Comments	Mandatory
destId	smallInt	The destinationId of the destination that you require categories and subcategories for. The destinationId can be retrieved from the /taxonomy/locations service.	N

### Example of use



### API Return Object

```
{
  data: [
    {
      id: 1,
      productCount: 47,
      thumbnailURL: "http://cache-graphicslib.viator.com/graphicslib-prelive/thumbs154x10",
      groupName: "Air, Helicopter & Balloon Tours",
      subcategories: [
```

```

    {
      categoryId: 1,
      subcategoryId: 1,
      subcategoryName: "Air Tours",
      sortOrder: 1
    },
    {
      categoryId: 1,
      subcategoryId: 3,
      subcategoryName: "Balloon Rides",
      sortOrder: 2
    },
    {
      categoryId: 1,
      subcategoryId: 2,
      subcategoryName: "Helicopter Tours",
      sortOrder: 3
    }
  ],
  sortOrder: 1
},
{
  id: 3,
  productCount: 4,
  thumbnailURL: "http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs154x100",
  groupName: "Cruises, Sailing & Water Tours",
  subcategories: [
    {
      categoryId: 3,
      subcategoryId: 7,
      subcategoryName: "Day Cruises",
      sortOrder: 1
    },
    {
      categoryId: 3,
      subcategoryId: 9,
      subcategoryName: "Night Cruises",
      sortOrder: 2
    }
  ],
  sortOrder: 2
},
{
  id: 4,
  productCount: 7,
  thumbnailURL: "http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs154x100",
  groupName: "Cultural & Theme Tours",
  subcategories: [
    {
      categoryId: 4,
      subcategoryId: 12,
      subcategoryName: "Historical & Heritage Tours",
      sortOrder: 1
    },
    {
      categoryId: 4,
      subcategoryId: 118,
      subcategoryName: "Ghost & Vampire Tours",
      sortOrder: 2
    },
    {
      categoryId: 4,
      subcategoryId: 13,
      subcategoryName: "Movie & TV Tours",
      sortOrder: 3
    }
  ]
}

```



```

    ],
    sortOrder: 3
  }
],
vmid: "321002",
errorMessage: null,
errorType: null,
dateStamp: "2012-09-03T17:42:59+0000",
errorReference: null,
errorMessageText: null,
success: true,
totalCount: 1,
errorName: null
}

```

### Description of elements in JSON Return object

Element	Type	Comments	To be viewed by consumer	Required field?
data	object	object which holds the category data	N	Y
id	smallint	The unique identifier for the category	N	Y
productCount	smallint	The number of products within the category, within the destination. This is null if a destination isn't passed.	N	Y
thumbnailURL		The URL to the thumbnail to use for the category. The image selected for the category is from the most popular tour. The value is null if a destination isn't passed.	Y	Y
groupName	varchar	The name of the group/category	Y	Y
subcategories	object	The object that holds the subcategories	N	Y
categoryId	smallint	The categoryId of the sub category	N	Y
subcategoryId	smallint	The identifier of the sub category within the category. This Id is unique within the category	N	Y
subcategoryName	varchar	The name of the sub category	Y	Y
sortOrder	smallint	In subcategories object: The suggested sort order of the sub category within the category. In data object: The suggested sort order of the category within the destination.	N	Y

## 5.3 Product APIs

### 5.3.1 /product

Method: GET



This API includes all the product detail that is seen on the website including information required for



price checks and booking such as agebands, tour grades, language options, booking questions, hotel pickup flags, etc.

### URL Parameters

code=2452T7&currencyCode=USD

### Description of URL parameters

Parameter	Type	Comments	Required field?
code	varchar	The product code of the activity/tour. The product code is available in the response of the search services in the "code" field	Yes
currencyCode	varchar	Set the currency of the product returned. The default currency is the currency of your account, or if you have multi-currency enabled, USD. Multi currency allows pricing and booking in various currencies - please speak to the BD team if you'd like this enabled. Note that you will be billed in the currency that the booking is made in.	No

### Example of use:





Go →

[Home](#) > [England](#) > [London](#) > **Tour**

## Leeds Castle Private Viewing, Canterbury and Greenwich Day Trip from London

Book Now →

Info
Reviews
Photos
Schedule

★★★★★ 97 reviews | [Add to Wishlist](#)



From USD  
**\$118.01**

### Details

Location: London, United Kingdom

Duration: 12 hours (Approx.)

Tour Code: 2452T7

### Highlights

- Full-day trip from London to Leeds Castle, White Cliffs of Dover, Canterbury and Greenwich
- Exclusive private viewing of Leeds Castle

### Example Result Object

```
{
  "errorReference": null,
  "data": {
    "supplierName": null,
    "currencyCode": "USD",
    "catIds": [
      5,
      4
    ],
    "subCatIds": [
      15,
      12
    ],
    "webURL": "http://prelive.partner.viator.com/en/1234/tours/London/Leeds-Castle-Private-Viewing-Canterbury-and-Greenwich-Day-Trip",
    "specialReservationDetails": null,
    "panoramaCount": 0,
    "merchantCancellable": false,
  }
}
```

```

"bookingEngineId": "FreesaleBE",
"onRequestPeriod": null,
"primaryGroupId": null,
"voucherRequirements": "You can present either a paper or an electronic voucher for thi
"tourGradesAvailable": true,
"hotelPickup": true,
"userPhotos": [
  {
    "sortOrder": 1,
    "ownerName": "Dianne S",
    "ownerCountry": "Australia",
    "productTitle": "Leeds Castle Private Viewing, Canterbury and Greenwich Day Tri
    "ownerAvatarURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/medi
    "thumbnailURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/media/
    "productCode": "2452T7",
    "caption": "<p>Greenwich</p>",
    "ownerId": 1329896,
    "editorsPick": false,
    "photoId": 5124815,
    "photoURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/media/cf/i
    "photoHiResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/medi
    "photoMediumResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/m
    "timeUploaded": "2012-09-21",
    "title": "IMG_1965"
  },
  {
    "sortOrder": 2,
    "ownerName": "Dianne S",
    "ownerCountry": "Australia",
    "productTitle": "Leeds Castle Private Viewing, Canterbury and Greenwich Day Tri
    "ownerAvatarURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/medi
    "thumbnailURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/media/
    "productCode": "2452T7",
    "caption": "<p>Canterbury Cathedral</p>",
    "ownerId": 1329896,
    "editorsPick": false,
    "photoId": 5124814,
    "photoURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/media/ce/i
    "photoHiResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/medi
    "photoMediumResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/m
    "timeUploaded": "2012-09-21",
    "title": "IMG_1935"
  },
  {
    "sortOrder": 3,
    "ownerName": "Dianne S",
    "ownerCountry": "Australia",
    "productTitle": "Leeds Castle Private Viewing, Canterbury and Greenwich Day Tri
    "ownerAvatarURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/medi
    "thumbnailURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/media/
    "productCode": "2452T7",
    "caption": "<p>Lunch Stop at Dover - Well worth the price</p>",
    "ownerId": 1329896,
    "editorsPick": false,
    "photoId": 5124813,
    "photoURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/media/cd/i
    "photoHiResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/medi
    "photoMediumResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/m
    "timeUploaded": "2012-09-21",
    "title": "IMG_1922"
  }
],
"reviews": [
  {
    "sortOrder": 1,
    "ownerName": "Jackie R",

```



```

        "ownerCountry": null,
        "productTitle": "Leeds Castle Private Viewing, Canterbury and Greenwich Day Tri",
        "ownerAvatarURL": null,
        "review": "<p>A wonderful day trip! All the stops are beautiful! Well paced.</p>",
        "rating": 5,
        "productCode": "2452T7",
        "publishedDate": "2014-08-03",
        "ownerId": 1225727,
        "viatorFeedback": "",
        "reviewId": 9643913,
        "submissionDate": "2014-08-03",
        "viatorNotes": ""
    },
    {
        "sortOrder": 2,
        "ownerName": "Philander C",
        "ownerCountry": null,
        "productTitle": "Leeds Castle Private Viewing, Canterbury and Greenwich Day Tri",
        "ownerAvatarURL": null,
        "review": "<p>Leeds Castle has an unique beauty all its own. Visiting it was on",
        "rating": 5,
        "productCode": "2452T7",
        "publishedDate": "2014-07-03",
        "ownerId": 8438717,
        "viatorFeedback": "",
        "reviewId": 9276742,
        "submissionDate": "2014-06-28",
        "viatorNotes": ""
    },
    {
        "sortOrder": 3,
        "ownerName": "Anton W",
        "ownerCountry": "USA",
        "productTitle": "Leeds Castle Private Viewing, Canterbury and Greenwich Day Tri",
        "ownerAvatarURL": null,
        "review": "<p>This was an excellent tour. Our guide was exceptionally well info",
        "rating": 5,
        "productCode": "2452T7",
        "publishedDate": "2014-06-20",
        "ownerId": 227091,
        "viatorFeedback": "",
        "reviewId": 9196860,
        "submissionDate": "2014-06-20",
        "viatorNotes": ""
    }
],
"videos": null,
"tourGrades": [
    {
        "sortOrder": 1,
        "currencyCode": "USD",
        "langServices": {
            "en/SERVICE_GUIDE": "English - Guide"
        },
        "priceFromFormatted": "$140.75",
        "gradeCode": "EN7",
        "priceFrom": 140.75,
        "merchantNetPriceFrom": 0,
        "merchantNetPriceFromFormatted": "",
        "gradeTitle": "Tour Only ",
        "gradeDepartureTime": "8:00 AM",
        "gradeDescription": "Tour led by an English-speaking guide",
        "defaultLanguageCode": "en"
    }
],
"ageBands": [

```



```

    {
      "sortOrder": 1,
      "ageFrom": 60,
      "ageTo": 99,
      "bandId": 3,
      "pluralDescription": "Seniors",
      "adult": false,
      "treatAsAdult": true,
      "description": "Senior",
      "count": 0
    },
    {
      "sortOrder": 2,
      "ageFrom": 16,
      "ageTo": 59,
      "bandId": 1,
      "pluralDescription": "Adults",
      "adult": true,
      "treatAsAdult": true,
      "description": "Adult",
      "count": 0
    },
    {
      "sortOrder": 3,
      "ageFrom": 3,
      "ageTo": 15,
      "bandId": 0,
      "pluralDescription": "Children",
      "adult": false,
      "treatAsAdult": false,
      "description": "Child",
      "count": 0
    },
    {
      "sortOrder": 4,
      "ageFrom": 0,
      "ageTo": 2,
      "bandId": 2,
      "pluralDescription": "Infants",
      "adult": false,
      "treatAsAdult": false,
      "description": "Infant",
      "count": 0
    }
  ],
  "bookingQuestions": [ ],
  "highlights": null,
  "salesPoints": [
    "Free snacks included",
    "Hotel pickup included",
    "Informative, friendly and professional guide",
    "Enjoy a day away from the hustle and bustle of the city",
    "Exclusive after-hours access",
    "Travel by boat and see the sights from the water"
  ],
  "ratingCounts": {
    "1": 9,
    "2": 22,
    "3": 27,
    "4": 86,
    "5": 107
  },
  "termsAndConditions": null,
  "voucherOption": "VOUCHER",
  "departureTime": "8:15am",
  "departureTimeComments": "<p>Hotel pickups commence prior to this time, you must contact

```

```

    "departurePoint": "Tour departs from Victoria Coach Station or select London hotels",
    "inclusions": [
        "Local guide",
        "Entrance fees",
        "Thames River cruise",
        "Hotel pickup (selected hotels only)",
        "Transport by air-conditioned coach"
    ],
    "itinerary": "",
    "destinationId": 737,
    "productPhotos": [
        {
            "photoURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/2452/SITou
            "caption": "Canterbury Cathedral, England",
            "supplier": ""
        }
    ],
    "city": "London",
    "additionalInfo": [
        "Confirmation will be received at time of booking",
        "Children under 16 years must be accompanied by an adult"
    ],
    "operates": "<strong> </strong><span><strong>April to October</strong><br>Daily, except
    "returnDetails": "Concludes at Embankment Pier at approximately 7pm",
    "mapURL": null,
    "specialOffer": "",
    "exclusions": [
        "Hotel drop-off",
        "Gratuities (optional)",
        "Food and drinks, unless specified"
    ],
    "location": "London, United Kingdom",
    "country": "United Kingdom",
    "region": "England-London",
    "description": "<div>Board a coach at the Victoria Coach Station or your London hotel a
    "shortDescription": "Spend a day touring the highlights of Kent on this day trip from L
    "price": 140.75,
    "rrp": 0,
    "specialReservation": false,
    "rating": 4.5,
    "thumbnailURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs154x109/
    "photoCount": 33,
    "reviewCount": 251,
    "supplierCode": "2452",
    "rrpformatted": "",
    "onSale": false,
    "savingAmount": 0,
    "savingAmountFormatted": "",
    "translationLevel": 0,
    "primaryDestinationId": 737,
    "primaryDestinationName": "London",
    "priceFormatted": "$140.75",
    "merchantNetPriceFrom": 0,
    "merchantNetPriceFromFormatted": "",
    "specialOfferAvailable": false,
    "thumbnailHiResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs674
    "shortTitle": "Leeds Castle Private Viewing, Canterbury and Greenwich Day Trip from Lon
    "videoCount": 0,
    "title": "Leeds Castle Private Viewing, Canterbury and Greenwich Day Trip from London",
    "duration": "11 hours",
    "code": "2452T7"
  },
  "dateStamp": "2014-12-03T20:11:39+0000",
  "errorType": null,
  "errorMessage": null,
  "errorName": null,

```



```

    "success": true,
    "totalCount": 1,
    "errorMessageText": null,
    "vmid": "321001"
  }

```

### Description of elements in JSON Response object

Object	Element	Type	Comments	To be viewed by consumer	Required field ?
<b>data</b>		object	main response object	N	Y
	location	varchar	location of the tour. Can be a combination of the country and city, or a custom location.	Y	Y
	country	varchar	The name of the country the product is based in	Y	Y
	description	medium text	Extended description of the product. May contain basic HTML mark-up tags: eg.  , <LI>, <B>, <U>, <P>, <I>, <UL>, <OL>, <FONT COLOR="#FF0000">	Y	Y
	departureTime	medium text	Descriptive information on departure times	Y	Y
	departureTime Comments	medium text	Extra comments in addition to the departure time information	Y	Y
	departurePoint	medium text	Departure point of tour	Y	Y
	inclusions	medium text	List of Inclusions in the product offering. May contain basic HTML mark-up tags: eg.  , <LI>, <B>, <U>, <P>, <I>, <UL>, <OL>	Y	Y
	itinerary	medium text	Itinerary of tour. May contain basic HTML mark-up tags: eg.  , <LI>, <B>, <U>, <P>, <I>, <UL>, <OL>	Y	Y
	VoucherRequirements	medium text	The description of the voucher requirements for this tour. Should reflect the value in voucherOption	Y	Y
	tourGradesAvailable	boolean	Flag to indicate if there are any available tour grades for this product. If this is true, you will need to display the tour grades to the consumer and pass a tour grade when booking this product. If this is false, only a default tour grade is available.	N	Y
	supplierName	varchar	not available to API partners	N	Y



	termsAndConditions	medium text	<p>The termsAndConditions field will return one of three values.</p> <p>1 - If the product has a standard/default policy, termsAndConditions will have a value of null.</p> <p>2 - Preset policy, which is "All sales are final and incur 100% cancellation penalties.&lt;br&gt;"</p> <p>3 - Custom policy, which is anything that is different to the values above.</p> <p>e.g.</p> <p>"If you cancel at least 8 day(s) in advance of the scheduled departure, there is a 10 per cent cancellation fee etc"</p> <p>The standard/default terms and conditions can be found at:</p> <p>Prelive - <a href="http://prelive.partner.viator.com/popups/terms.jspa#payments">http://prelive.partner.viator.com/popups/terms.jspa#payments</a></p> <p>Live - <a href="http://www.partner.viator.com/popups/terms.jspa#payments">http://www.partner.viator.com/popups/terms.jspa#payments</a></p> <p>Which the API partner can link to.</p> <p>See section on <a href="#">Support APIs</a>.</p>	Y	Y
	voucherOption	varchar	<p>VOUCHER_PAPER_ONLY - Paper Vouchers only accepted</p> <p>VOUCHER_E - EVouchers + Paper Vouchers accepted</p> <p>VOUCHER_ID_ONLY - ID + Evouchers + Paper vouchers accepted</p>	Y	Y
	destinationId	smallint	Unique id of the destination of the product. The list of destinationIds is available in /taxonomy/locations	N	Y
	city	varchar	destination / city name that the product operates in.	Y	Y
	exclusions	medium text	Exclusions from the product offering. May contain basic HTML mark-up tags: eg.  , <LI>, <B>, <U>, <P>, <I>, <UL>, <OL>	Y	Y
	region	varchar	Name of the region that the product operates in.	Optional	Y
	operates	medium text	Frequency of the tour. e.g. Daily. May contain basic HTML mark-up tags such as <p>	Optional	Y
	returnDetails	varchar	Details of the tour drop-off	Y	Y
	highlights	medium text	Not available to API partners. Use salesPoints instead.	N	Y
	salesPoints	medium text	Selling points of the tour. Available on most products.	Y	Y
	mapURL	varchar	URL to the map of the tour, if available. Usually an image	N	Y
	hotelPickup	boolean	Flag to indicate if the tour offers hotel pick up. If this is true, you will need to collect hotel information	Y	Y
	specialOffer	medium text	If the product has a special offer, the details of the special itself. Will be an empty string if there are no specials available.	Y	Y
	duration		Description of the duration of the tour	Y	Y



	shortDescription	medium text	shortened description of the tour	Optional	Y
	code	varchar	Product code	Y	Y
	supplierCode	varchar	The unique identifier for suppliers	N	Y
	title	varchar	title of the the product	Y	Y
	catIds	smallint	List of categoryIds that the product falls in. The list of categoryIds available for the destination is available in /taxonomy/categories	N	Y
	subCatIds	smallint	List of subcategoryIds that the product falls in. The subcategoryIds are listed in /taxonomy/categories.		
	thumbnailURL	varchar	URL of the image thumbnail of the product.	Y	Y
	photoCount	int	The number of user photos published for the product	Optional	Y
	reviewCount	int	The number of reviews published for the product	Optional	Y
	specialReservation	boolean	Viator only. Will always be false for partners.	N	Y
	rrp	decimal	Comparison price of the product on another system, website or company.	N	Y
	rrpformatted	varchar	The comparison price, formatted to two decimal places if applicable and with the currency symbol	Y	Y
	savingAmount	decimal	If there is a discount on the product's comparison price (rrp), the savings amount will be displayed. If there are no savings, the value will be 0	N	Y
	savingAmountFormatted	varchar	The discount given on this product compared to the comparison price (rrp). If there are no savings amount, the value will be blank.	Y - if any savings	Y
	webURL	varchar	URL of the product on the API partner's white label site.	Optional	Y
	videoCount	int	The number of videos associated to the product	N	Y
	panoramaCount	int	The number of panoramas associated to the product	Optional	Y
	merchantCancellable	boolean	This is applicable to merchant partners only. This value will always be false for retail API partners.	No	Y
	shortTitle	varchar	The shortened version of the title, for use by mobile devices	Optional	Y
	translationLevel	int	A number indicating the amount of text translated for a non-English product. 0 - No translation (English Only) 20 - Only the product's ProductName and ProductText have been translated 90 - Full machine translation 100 - Full human translation	No	Y
	primaryDestinationId	smallint	The destinationId of the product's primary destination	N	Y
	primaryDestinationName	varchar	The name of the product's primary destination	N	Y
	price	decimal	This is "from" price of the tour. This is the cheapest price, taking into consideration off peak periods and discounts on larger groups.	N	Y

	priceFormatted	varchar	The price of the product formatted, with two decimal places if applicable, and the currency symbol.	Y	Y
	merchantNetPriceFrom	decimal	The merchant net price from, in decimal format. This will always be 0 for retail API partners.		
	merchantNetPriceFromFormatted	varchar	the merchant net price from, in the currency selected and formatted according to the currency. It will always be a blank value "" for retail API partners.		
	onSale	boolean	Indicates if a discount is applied to the price	N	Y
	specialOfferAvailable	boolean	Indicates if a special offer is available for the product	N	Y
	thumbnailURL HiResURL	varchar	URL of the high resolution thumbnail for the product	optional	Y
	specialReservationDetails	medium text	Viator only. Will be null for partners.	N	Y
	additionalInfo	medium text	Additional information on the Product offering. May contain basic HTML mark-up tags: eg.  , <LI>, <B>, <U>, <P>, <I>, <UL>, <OL>	Y	Y
	bookingEngineId	varchar	Indicates whether the booking will be confirmed immediately or if it will be pending after the booking has been made.  FreesaleBE – The product will be confirmed immediately and the supplier will be sent a notification. UnconditionalBE - The product will be confirmed immediately and the supplier will not be notified. DeferredCRMBE - The product is an on request product and will not be confirmed immediately. The product will go to pending after it is booked, to be confirmed by the supplier within the time specified in the hoursConfirmed field available in the booking response and post-booking services. FreesaleOnRequestBE - The product is freesale up until a specific number of days before the travel date. The specific number of days is referred to as the on request period. If a booking is made within the on request period, the product will be an On Request product. Once the booking has been made, the bookingEngineId will change to either FreesaleOnRequestBE:OnRequest or FreesaleOnRequestBE:Freesold depending on the travel date and the on request period.	N	Y
	onRequestPeriod	int	This will contain a value if the bookingEngineId is FreesaleOnRequestBE. This represents the number of hours before the travel date that the product will be on request for. For instance, a product with an onRequestPeriod of 48 hours means that the product is freesale up until 48 hours before the travel date, and is on request for 48 hours or less until the travel date (note that "hours in advance" may also come into play - hours in advance is the number of hours a product is available for booking before the travel date. This is	N	Y

			not available in the API)		
<b>ageBands</b>		object	Lists the agebands available for this product. Only the agebands available can book this tour.	N	Y
	description		The ageBands available in the response object can be a combination of the descriptions below: Adult Child Infant Youth Senior	Y	Y
	count	int	Not used	N	Y
	ageFrom	int	The start of the age range, from (and including) this age.	Optional	Y
	ageTo	int	The end of the range, from (and including) this age.	Optional	Y
	adult	boolean	Indicates which ageband is the adult ageband.	N	Y
	bandId	int	The id of the ageband.	N	Y
	pluralDescription	medium text	The plural of the ageband.	Optional	Y
	treatAsAdult	boolean	The leadtraveller must have treatAsAdult as true. When the user is selecting the passenger mix, the bandId of the lead traveller must have treatAsAdult as true.	N	Y
	sortOrder	smallint	The order the ageband should be displayed in	N	Y
<b>tourGrades</b>		object	The tourgrades / option available in this tour. A product can have many tourgrades. Tourgrades can represent a different time or an additional feature, to name a few.	N	Y
	currencyCode	varchar	Currency code of the tour grade	Optional	Y
	langServices	list	list of language services codes available for the tour grade in different languages. Format is <languagecode>/<service> : <description of service> service: SERVICE_AUDIO, SERVICE_WRITTEN, SERVICE_GUIDE Returns null if no language services are available. If any language services are available, the languagecode/service must be passed to /booking/book.	Y	Y
	gradeCode	varchar	The unique code for the tour grade. This code is only unique within the product.	Optional	Y
	PriceFrom	decimal	The starting price of the tour grade.	N	Y
	priceFromFormatted	varchar	The starting price of the tour grade, including currency symbol and decimal places.	Y	Y
	gradeTitle	varchar	Title of the tour grade	Y	Y
	gradeDepartureTime	varchar	The times the tour grade departs. Usually includes AM/PM . If the gradeDepartureTime is not available, it may be because it is available all day or already covered by the gradeDescription	Y	Y
	gradeDescription	varchar	Description of the tour grade	Y	Y

	defaultLanguageCode	varchar	Default language code of the tour grade	Y	Y
<b>videos</b>		object	Array of videos asociated to the product	N	Y
	private	boolean	Flag to indicate if it should be displayed on the front end	N	N
	source	enum	YOUTUBE	Optional	N
	description	varchar	Description of the video	Y	N
	shortDescription	varchar	A shortened version of the description	Optional	N
	productCode	varchar	product code the video is associated to	Optional	N
	title	varchar	Title of video	Y	N
	thumbnailURL	varchar	URL of thumbnail for the video	Optional	N
	ownerId	int	user account Id. -1 indicates a Viator submitted photo.	Optional	N
	videoId	varchar	Id of video in source	Optional	N
	copyright	varchar	Copyright text	Optional	N
	photoHiResURL	varchar	High resolution version of the photo for the video	Optional	N
	productTitle	varchar	Product title of video	Optional	N
	ownerName	varchar	Name of the user who submitted the video	Optional	N
	ownerCountry	varchar	Country code of the user that submitted the video. Null indicates a Viator submitted video.	Optional	N
	ownerAvatarURL	varchar	Profile image of the user who submitted the video.	Optional	N
	sortOrder	smallint	Order to display the video	N	N
<b>productPhotos</b>		object	Product photos can be supplied by the tour provider.	N	Y
	photoURL	varchar	URL of the photo to display	Y	N
	caption	varchar	Caption of the product photo	Y	N
	supplier	varchar	The name of the supplier	Optional	N
<b>reviews</b>		object	Array of reviews submitted for this product. Reviews have a submitted date and a published date (approved by Viator).	N	Y
	productCode	varchar	The product code the review was submitted for.	Optional	N
	review	medium text	The review text	Y	N
	rating	int	The rating the user gave the tour. This can be represented on the front end using images	Y	N
	viatorFeedback	medium text	Any feedback returned from viator on the review submitted	Optional	N
	reviewId	int	Id of the review submitted	N	N

	submissionDate	date	Date the review was submitted	Optional	N
	publishedDate	date	Date the review was approved by Viator to be displayed on the front end	Optional	N
	viatorNotes	medium text	Any additional notes from viator regarding the review	Optional	N
	productTitle	varchar	Title of the product	Optional	N
	ownerName	varchar	Name of the user who submitted the review	Y	N
	ownerCountry	varchar	Country code of the user that submitted the review	optional	N
	ownerAvatarURL	varchar	Profile image of the user who submitted the review	optional	N
	sortOrder	smallint	Sort order of the review	N	N
<b>userPhotos</b>		object	Array of photos submitted by users. Different resolutions of the photo are available. Thumbnail can be displayed in a gallery, and the medium/high res version can be displayed when the user clicks on it.	N	Y
	productCode	varchar	The product code the photo was submitted for	Optional	N
	title	varchar	title of the photo submitted	Y	N
	caption	varchar	caption of the photo submitted	Y	N
	thumbnailURL	varchar	URL of the thumbnail of the photo	Optional	N
	ownerId	int	user account Id	N	N
	editorsPick	boolean	Indicates if the photo has been selected as an editors' pick	Optional	N
	photoHiResURL	varchar	URL of the high resolution version of the photo	Optional	N
	photoId	int	Id of the photo	N	N
	photoURL	varchar	URL of the photo	Optional	N
	photoMediumResURL	varchar	URL of the medium resolution version of the photo	Optional	N
	timeUploaded	timestamp	Timestamp of when the photo was uploaded	Optional	N
	productTitle	varchar	Title of the product	Optional	N
	ownerName	varchar	Name of the user who submitted the photo	Y	N
	ownerCountry	char	Country code of the user that submitted the photo	Optional	N
	ownerAvatarURL	varchar	Profile image of the user who submitted the photo	Optional	N
	sortOrder	smallint	Sort order of the photo when displaying the photos	N	N
<b>bookingQuestions</b>		object	Additional information needed for the product will be included in the bookingQuestions object. The questions should be displayed to the customer with the title, message, subtitle and whether it is mandatory (required) or not.	Y	Y

	message	varchar	The question to display to the customer.	Y - on check out	N
	required	boolean	Indicates if an answer must be passed when booking the product	Y - on check out	N
	title	varchar	The title of the question	Y - on check out	
	questionId	varchar	An unique ID given to each question within a category/subcategory. Each question represents a special requirement on the checkout page.	Y - on check out	N
	subTitle	varchar	An example of what format the answer should be written in.	Y - on check out	N
	sortOrder	smallint	Order that the question should be displayed in on the checkout page.	Y - on check out	N
<b>ratingCounts</b>		object	Breakdown of the star ratings for the product	N	Y
	1	smallint	The number of 1 star reviews for the product	Optional	Y
	2	smallint	The number of 2 star reviews for the product	Optional	Y
	3	smallint	The number of 3 star reviews for the product	Optional	Y
	4	smallint	The number of 4 star reviews for the product	Optional	Y
	5	smallint	The number of 5 star reviews for the product	Optional	Y

### 5.3.2 /product/reviews

Method: GET

#### Sample Query String

code=2280AAHT&topX=1-5&sortOrder=REVIEW\_RATING\_D

#### Valid sort orders:

REVIEW\_RATING\_A - Traveler Rating (low->high) (not averages)  
 REVIEW\_RATING\_D - Traveler Rating (high->low) (not averages)  
 REVIEW\_RATING\_SUBMISSION\_DATE\_D - Most recent review





[Home](#) > [England](#) > [London](#) > [Tour Reviews](#)

## Leeds Castle Private Viewing, Canterbury and Greenwich Day Trip from London

Sort by:

★★★★★

Totally enjoyed this tour as compared to my previous one from another company... I had a very knowledgeable guide - Paul... He's a walking encyclopedia, almost everything we walked past were explained in depth... Well done!

*Reviewed by Connie Sue Ling Y, Singapore, January 2011*

★★★★★

Great sights!!

*Reviewed by Chris L, New Zealand, January 2011*

★★★★★

great guide. he was very passionate about the sites were seeing. Made some quite decisions as to what to do when the weather became not so good. Paul's good decision that was

### Example Result Object

```
{
  "data": [
    {
      "productCode": "2280AAHT",
      "review": "Magnificent and breathtaking. Enjoyed every minute.",
      "rating": 5,
      "ownerId": 30003,
      "viatorFeedback": "",
      "reviewId": 501661,
      "submissionDate": "2006-10-21",
      "publishedDate": "2006-11-08",
      "viatorNotes": "",
      "productTitle": "Grand Canyon All American Helicopter Tour",
      "ownerName": "Diane B",
      "ownerCountry": "USA",
      "ownerAvatarURL": null,
      "sortOrder": 1
    },
  ],
}
```



```

        "productCode": "2280AAHT",
        "review": "Fantastic! Excellent service, great pilot, incredible scenery. Thanks ve
        "rating": 5,
        "ownerId": 30170,
        "viatorFeedback": "",
        "reviewId": 501710,
        "submissionDate": "2006-10-31",
        "publishedDate": "2006-11-08",
        "viatorNotes": "",
        "productTitle": "Grand Canyon All American Helicopter Tour",
        "ownerName": "Martin C",
        "ownerCountry": "Canada",
        "ownerAvatarURL": null,
        "sortOrder": 2
    },
    {
        "productCode": "2280AAHT",
        "review": "What a trip, it all began with a limo from the hotel. The flight out to
        "rating": 5,
        "ownerId": 30615,
        "viatorFeedback": "",
        "reviewId": 501857,
        "submissionDate": "2006-11-20",
        "publishedDate": "2006-11-21",
        "viatorNotes": "",
        "productTitle": "Grand Canyon All American Helicopter Tour",
        "ownerName": "Akinfemi S",
        "ownerCountry": "United Kingdom",
        "ownerAvatarURL": null,
        "sortOrder": 3
    },
    {
        "productCode": "2280AAHT",
        "review": "A lot of fun. I loved the music and great commentary. Not a cheap trip b
        "rating": 5,
        "ownerId": 30912,
        "viatorFeedback": "",
        "reviewId": 501947,
        "submissionDate": "2006-10-23",
        "publishedDate": "2006-11-08",
        "viatorNotes": "",
        "productTitle": "Grand Canyon All American Helicopter Tour",
        "ownerName": "Steven P",
        "ownerCountry": "USA",
        "ownerAvatarURL": null,
        "sortOrder": 4
    },
    {
        "productCode": "2280AAHT",
        "review": "We were totally impressed with the ease in online booking, telephone con
        "rating": 5,
        "ownerId": 33469,
        "viatorFeedback": "",
        "reviewId": 502760,
        "submissionDate": "2006-11-17",
        "publishedDate": "2006-11-21",
        "viatorNotes": "",
        "productTitle": "Grand Canyon All American Helicopter Tour",
        "ownerName": "Janet P",
        "ownerCountry": "Canada",
        "ownerAvatarURL": "http://qa.graphicslib.viator.com/graphicslib-prelive/media/d5/da
        "sortOrder": 5
    }
},
"vmid": "221001",
"errorMessage": null,

```





```
"errorType": null,  
"dateStamp": "2012-09-03T14:34:02+0000",  
"errorReference": null,  
"errorMessageText": null,  
"success": true,  
"totalCount": 1575,  
"errorName": null  
}
```

### 5.3.3 /product/photos

Method: GET

Get a products USER photos

#### Sample Query String

code=2280AAHT&topX=4-10

#### HTTPS images

For sites requiring https images, parse out <http://cache.graphicslib.viator.com/> for <https://graphicslib.viator.com/> from the photo URL.

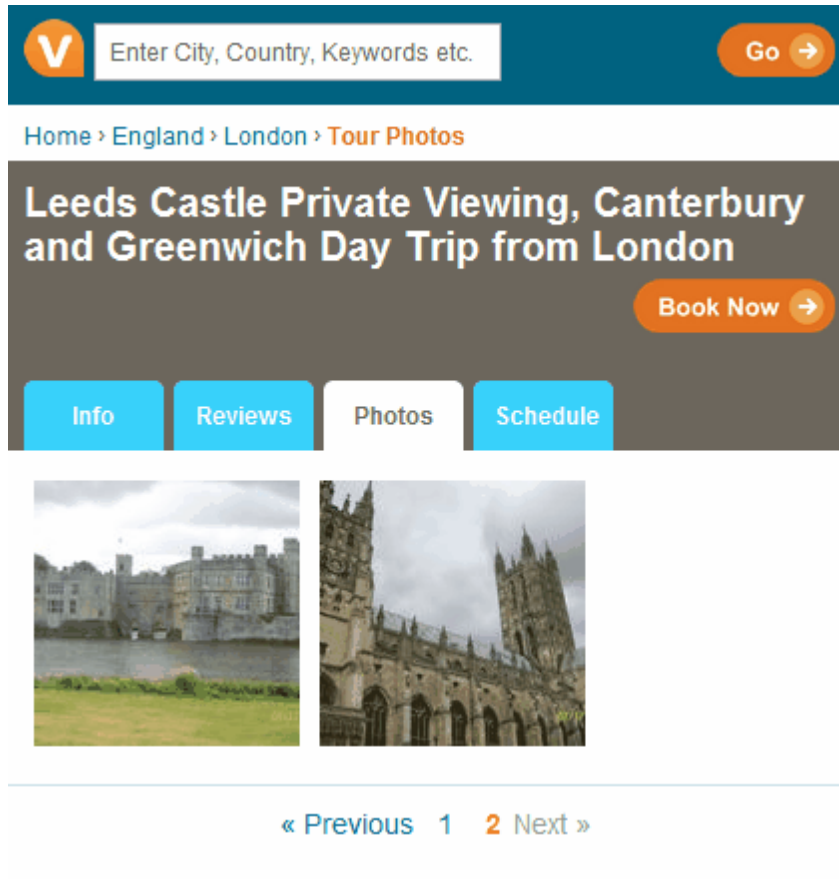
e.g.

"thumbnailHiResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs674x446/3422/SITours/las-vegas-night-strip-helicopter-tour-in-las-vegas-120300.jpg"

The https version of this image is:

<https://graphicslib.viator.com/graphicslib-prelive/thumbs674x446/3422/SITours/las-vegas-night-strip-helicopter-tour-in-las-vegas-120300.jpg>





### Example Result Object

```
{
  data: [
    {
      productCode: "2280AAHT",
      title: "sphinx-luxor-egypt.jpg",
      caption: "added in VCS aug 1 editors pick",
      thumbnailURL: "http://qa.graphicslib.viator.com/graphicslib-prelive/media/fd/sphinx-luxor-egypt.jpg",
      ownerId: 1215729,
      editorsPick: true,
      photoId: 1215741,
      photoURL: "http://qa.graphicslib.viator.com/graphicslib-prelive/media/fd/sphinx-luxor-egypt.jpg",
      photoHiResURL: "http://qa.graphicslib.viator.com/graphicslib-prelive/media/fd/sphinx-luxor-egypt.jpg",
      photoMediumResURL: "http://qa.graphicslib.viator.com/graphicslib-prelive/media/fd/sphinx-luxor-egypt.jpg",
      timeUploaded: "2011-08-01",
      productTitle: "Grand Canyon All American Helicopter Tour",
      ownerName: "Daveo",
      ownerCountry: "Australia",
      ownerAvatarURL: "http://qa.graphicslib.viator.com/graphicslib-prelive/media/f1/daveo.jpg",
      sortOrder: 1
    },
    {
      productCode: "2280AAHT",
      title: "Egypt022.JPG",
      caption: "added in VCS aug 1 editors pick",
      thumbnailURL: "http://qa.graphicslib.viator.com/graphicslib-prelive/media/fc/egypt022.jpg",
      ownerId: 1215729,
      editorsPick: true,
      photoId: 1215740,
      photoURL: "http://qa.graphicslib.viator.com/graphicslib-prelive/media/fc/egypt022.jpg",
      photoHiResURL: "http://qa.graphicslib.viator.com/graphicslib-prelive/media/fc/egypt022.jpg"
    }
  ]
}
```

```

        photoMediumResURL: "http://qa.graphicslib.viator.com/graphicslib-prelive/media/fc/
        timeUploaded: "2011-08-01",
        productTitle: "Grand Canyon All American Helicopter Tour",
        ownerName: "Daveo",
        ownerCountry: "Australia",
        ownerAvatarURL: "http://qa.graphicslib.viator.com/graphicslib-prelive/media/f1/dav
        sortOrder: 2
    },
    {
        productCode: "2280AAHT",
        title: "Grand Canyon",
        caption: "View from inside the canyon",
        thumbnailURL: "http://qa.graphicslib.viator.com/graphicslib-prelive/media/4b/grand
        ownerId: 1106473,
        editorsPick: true,
        photoId: 1165131,
        photoURL: "http://qa.graphicslib.viator.com/graphicslib-prelive/media/4b/grand-can
        photoHiResURL: "http://qa.graphicslib.viator.com/graphicslib-prelive/media/4b/gran
        photoMediumResURL: "http://qa.graphicslib.viator.com/graphicslib-prelive/media/4b/
        timeUploaded: "2010-12-02",
        productTitle: "Grand Canyon All American Helicopter Tour",
        ownerName: "GARY I",
        ownerCountry: null,
        ownerAvatarURL: null,
        sortOrder: 3
    }
],
    vmid: "221001",
    errorMessage: null,
    errorType: null,
    dateStamp: "2012-06-18T16:59:16+0000",
    errorReference: null,
    errorMessageText: null,
    success: true,
    totalCount: 377,
    errorName: null
}

```

## 5.4 Search APIs

Search APIs include search for products by date and destination, getting a list of products by code or searching with freetext which can include different types of results.

### 5.4.1 /search/products

Method: POST

This returns a list of products

Dates are optional.

Destination ID and currencyCode are required.

#### JSON Request Parameters

```

{
    "startDate": "2015-10-25",
    "endDate": "2015-10-27",
    "topX": "1-5",
    "destId": 684,
    "currencyCode": "EUR",
    "catId": 0,
    "subCatId": 0,
    "dealsOnly": false,
    "sortOrder": "PRICE_FROM_A"
}

```



```
}
```

**sortOrder**

Valid sort orders:

- TOP\_SELLERS - The top sellers - default sortOrder
- REVIEW\_AVG\_RATING\_A - Traveler Rating (low->high) Average
- REVIEW\_AVG\_RATING\_D - Traveler Rating (high->low) Average
- PRICE\_FROM\_A - Price (low->high)
- PRICE\_FROM\_D - Price (high->low)

**topX**

For returning results in particular positions. e.g. a value of 11-15 will return the results in position 11 to 15 based on the sort (total of 5 products). Useful for pagination.

**catId**

Category id from the /service/taxonomy/categories

**subCatId**


Sub Category id from the /service/taxonomy/categories

**dealsOnly**

Applicable only when there is no startDate/endDate as it is a destination search only feature.

This API is used for obtaining lists of products for destination pages.

**Sample Destination Product List:**



[Home](#) > [England](#) > [London](#)


## London

### TOURS & THINGS TO DO

[Things To Do](#)
[Deals](#)
[Attractions](#)

Sort by:

### Skip the Line: Buckingham Palace Tour



★★★★★


262 reviews

From USD

**\$28.68**

Check Dates

### London Pass



★★★★★

161 reviews

From USD

**\$63.92**

Check Dates

#### Example Result Object

```
{
  "errorReference": null,
  "data": [
    {
      "sortOrder": 1,
      "supplierName": null,
      "currencyCode": "USD",
      "catIds": [
        15
      ],
      "subCatIds": [
        101
      ],
      "webURL": "http://prelive.partner.viator.com/en/1234/tours/Las-Vegas/Las-Vegas-Mono",
      "specialReservationDetails": null,
      "panoramaCount": 0,
      "merchantCancellable": false,
      "bookingEngineId": "FreesaleBE",

```



```

        "onRequestPeriod": null,
        "primaryGroupId": 15,
        "title": "Las Vegas Monorail Ticket",
        "shortDescription": "See all of the Las Vegas Strip in a fast, easy, convenient way",
        "price": 11.99,
        "specialReservation": false,
        "rating": 4,
        "thumbnailURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs154x",
        "photoCount": 0,
        "reviewCount": 95,
        "supplierCode": "5859",
        "rrp": 0,
        "rrpFormatted": "",
        "savingAmount": 0,
        "savingAmountFormatted": "",
        "videoCount": 0,
        "onSale": false,
        "shortTitle": "Las Vegas Monorail Tickets",
        "translationLevel": 0,
        "primaryDestinationId": 684,
        "primaryDestinationName": "Las Vegas",
        "priceFormatted": "$11.99",
        "merchantNetPriceFrom": 0,
        "merchantNetPriceFromFormatted": "",
        "specialOfferAvailable": false,
        "thumbnailHiResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumb",
        "duration": "Varies",
        "code": "5859LVMON"
    },
    {
        "sortOrder": 2,
        "supplierName": null,
        "currencyCode": "USD",
        "catIds": [
            15
        ],
        "subCatIds": [
            52
        ],
        "webURL": "http://prelive.partner.viator.com/en/1234/tours/Las-Vegas/Las-Vegas-Airp",
        "specialReservationDetails": null,
        "panoramaCount": 0,
        "merchantCancellable": false,
        "bookingEngineId": "FreesaleBE",
        "onRequestPeriod": null,
        "primaryGroupId": 15,
        "title": "Las Vegas Airport Roundtrip Transfer",
        "shortDescription": "Get on the fast track to the Las Vegas Strip with a Las Vegas",
        "price": 14.49,
        "specialReservation": false,
        "rating": 4.5,
        "thumbnailURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs154x",
        "photoCount": 8,
        "reviewCount": 2424,
        "supplierCode": "5378",
        "rrp": 0,
        "rrpFormatted": "",
        "savingAmount": 0,
        "savingAmountFormatted": "",
        "videoCount": 0,
        "onSale": false,
        "shortTitle": "Las Vegas Airport Roundtrip Transfer",
        "translationLevel": 0,
        "primaryDestinationId": 684,
        "primaryDestinationName": "Las Vegas",
        "priceFormatted": "$14.49",

```

```
"merchantNetPriceFrom": 0,
"merchantNetPriceFromFormatted": "",
"specialOfferAvailable": false,
"thumbnailHiResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumb
",
"duration": "Varies",
"code": "5378LASAPTRND"
},
{
  "sortOrder": 3,
  "supplierName": null,
  "currencyCode": "USD",
  "catIds": [
    15
  ],
  "subCatIds": [
    52
  ],
  "webURL": "http://prelive.partner.viator.com/en/1234/tours/Las-Vegas/Private-Las-Ve
",
  "specialReservationDetails": null,
  "panoramaCount": 0,
  "merchantCancellable": false,
  "bookingEngineId": "FreesaleBE",
  "onRequestPeriod": null,
  "primaryGroupId": 15,
  "title": "Private Las Vegas Hotel to Airport Luxury Limousine Transfer",
  "shortDescription": "Live in the lap of luxury until the moment you leave Las Vegas
",
  "price": 16.59,
  "specialReservation": false,
  "rating": 4.5,
  "thumbnailURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs154x
",
  "photoCount": 4,
  "reviewCount": 54,
  "supplierCode": "2596",
  "rrp": 0,
  "rrpformatted": "",
  "savingAmount": 0,
  "savingAmountFormatted": "",
  "videoCount": 0,
  "onSale": false,
  "shortTitle": "Private Las Vegas Hotel to Airport Luxury Limousine Transfer",
  "translationLevel": 0,
  "primaryDestinationId": 684,
  "primaryDestinationName": "Las Vegas",
  "priceFormatted": "$16.59",
  "merchantNetPriceFrom": 0,
  "merchantNetPriceFromFormatted": "",
  "specialOfferAvailable": false,
  "thumbnailHiResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumb
",
  "duration": "15-45 minutes",
  "code": "2596LASHTLAPTPTVT"
},
{
  "sortOrder": 4,
  "supplierName": null,
  "currencyCode": "USD",
  "catIds": [
    8,
    21
  ],
  "subCatIds": [
    29,
    78
  ],
  "webURL": "http://prelive.partner.viator.com/en/1234/tours/Las-Vegas/Shark-Reef-at-
",
  "specialReservationDetails": null,
  "panoramaCount": 0,
```



```

    "merchantCancellable": false,
    "bookingEngineId": "FreesaleBE",
    "onRequestPeriod": null,
    "primaryGroupId": 8,
    "title": "Shark Reef at Mandalay Bay Hotel and Casino",
    "shortDescription": "Marvel at the incredible Shark Reef Aquarium in the heart of t
    "price": 18.49,
    "specialReservation": false,
    "rating": 0,
    "thumbnailURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs154x
    "photoCount": 0,
    "reviewCount": 0,
    "supplierCode": "5156",
    "rrp": 0,
    "rrpformatted": "",
    "savingAmount": 0,
    "savingAmountFormatted": "",
    "videoCount": 0,
    "onSale": false,
    "shortTitle": "Shark Reef at Mandalay Bay Hotel and Casino",
    "translationLevel": 0,
    "primaryDestinationId": 684,
    "primaryDestinationName": "Las Vegas",
    "priceFormatted": "$18.49",
    "merchantNetPriceFrom": 0,
    "merchantNetPriceFromFormatted": "",
    "specialOfferAvailable": false,
    "thumbnailHiResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumb
    "duration": "Varies",
    "code": "5156SHARK"
  },
  {
    "sortOrder": 5,
    "supplierName": null,
    "currencyCode": "USD",
    "catIds": [
      8
    ],
    "subCatIds": [
      31,
      29
    ],
    "webURL": "http://prelive.partner.viator.com/en/1234/tours/Las-Vegas/The-Mob-Museum
    "specialReservationDetails": null,
    "panoramaCount": 0,
    "merchantCancellable": false,
    "bookingEngineId": "FreesaleBE",
    "onRequestPeriod": null,
    "primaryGroupId": 8,
    "title": "The Mob Museum Admission",
    "shortDescription": "Immerse yourself in the gripping and gruesome stories of the m
    "price": 19.89,
    "specialReservation": false,
    "rating": 4.5,
    "thumbnailURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs154x
    "photoCount": 57,
    "reviewCount": 36,
    "supplierCode": "5932",
    "rrp": 0,
    "rrpformatted": "",
    "savingAmount": 0,
    "savingAmountFormatted": "",
    "videoCount": 0,
    "onSale": false,
    "shortTitle": "The Mob Museum Admission",
    "translationLevel": 0,

```





```

        "primaryDestinationId": 684,
        "primaryDestinationName": "Las Vegas",
        "priceFormatted": "$19.89",
        "merchantNetPriceFrom": 0,
        "merchantNetPriceFromFormatted": "",
        "specialOfferAvailable": false,
        "thumbnailHiResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumb",
        "duration": "Varies",
        "code": "5932MOB"
    }
],
"dateStamp": "2014-12-03T20:14:25+0000",
"errorType": null,
"errorMessage": null,
"errorName": null,
"success": true,
"totalCount": 160,
"errorMessageText": null,
"vmid": "321002"
}

```

### 5.4.2 /search/products/codes

Method: POST

This API returns an array of products given an array of product IDs.

Useful for wishlist display, myAccount, etc.

Note that an inactive / non-existent product code will return zero, null and blank values (as per the example below).

#### JSON Request Parameters

```

{
  "currencyCode": "EUR",
  "productCodes": ["2280SUN", "3333SYDOWN"]
}

```

#### Example Result Object

```

{
  "errorReference": null,
  "data": [
    {
      "sortOrder": 1,
      "supplierName": null,
      "currencyCode": "USD",
      "catIds": [
        1
      ],
      "subCatIds": [
        2
      ],
      "webURL": "http://prelive.partner.viator.com/en/1234/tours/Las-Vegas/Grand-Canyon-W",
      "specialReservationDetails": null,
      "panoramaCount": 0,
      "merchantCancellable": false,
      "bookingEngineId": "FreesaleOnRequestBE",
      "onRequestPeriod": 336,
      "primaryGroupId": null,
      "title": "Grand Canyon West Rim Deluxe Sunset Helicopter Tour",

```



```

    "shortDescription": "Take off from Las Vegas on a magical sunset helicopter tour to",
    "price": 489.99,
    "specialReservation": false,
    "rating": 5,
    "thumbnailURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs154x",
    "photoCount": 109,
    "reviewCount": 316,
    "supplierCode": "2280",
    "rrp": 0,
    "rrpformatted": "",
    "savingAmount": 0,
    "savingAmountFormatted": "",
    "videoCount": 0,
    "onSale": false,
    "shortTitle": "Grand Canyon West Rim Deluxe Sunset Helicopter Tour",
    "translationLevel": 0,
    "primaryDestinationId": 684,
    "primaryDestinationName": "Las Vegas",
    "priceFormatted": "$489.99",
    "merchantNetPriceFrom": 0,
    "merchantNetPriceFromFormatted": "",
    "specialOfferAvailable": false,
    "thumbnailHiResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumb",
    "duration": "3 hours 30 minutes",
    "code": "2280SUN"
  },
  {
    "sortOrder": 2,
    "supplierName": null,
    "currencyCode": "USD",
    "catIds": [
      8,
      21
    ],
    "subCatIds": [
      29,
      78
    ],
    "webURL": "http://prelive.partner.viator.com/en/1234/tours/Sydney/Manly-SEA-LIFE-Sa",
    "specialReservationDetails": null,
    "panoramaCount": 0,
    "merchantCancellable": false,
    "bookingEngineId": "UnconditionalBE",
    "onRequestPeriod": null,
    "primaryGroupId": null,
    "title": "Manly SEA LIFE Sanctuary Entrance Ticket",
    "shortDescription": "Enjoy a fun filled day out, the whole family can enjoy Manly S",
    "price": 15.03,
    "specialReservation": false,
    "rating": 2.5,
    "thumbnailURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs154x",
    "photoCount": 0,
    "reviewCount": 2,
    "supplierCode": "3333",
    "rrp": 0,
    "rrpformatted": "",
    "savingAmount": 0,
    "savingAmountFormatted": "",
    "videoCount": 0,
    "onSale": false,
    "shortTitle": "Manly SEA LIFE Sanctuary Entrance Ticket",
    "translationLevel": 0,
    "primaryDestinationId": 357,
    "primaryDestinationName": "Sydney",
    "priceFormatted": "$15.03",
    "merchantNetPriceFrom": 0,

```

```

        "merchantNetPriceFromFormatted": "",
        "specialOfferAvailable": false,
        "thumbnailHiResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumb
        "duration": "At leisure",
        "code": "3333SYDOWNM"
    }
},
"dateStamp": "2014-12-03T20:14:57+0000",
"errorType": null,
"errorMessage": null,
"errorName": null,
"success": true,
"totalCount": 2,
"errorMessageText": null,
"vmid": "321002"
}

```

### 5.4.3 /search/freetext

Method: POST

This API provides a free text search across all products, destinations, attractions and recommendations.

The text parameter is required.

#### JSON Request Parameters

```

{
  "topX": "2-6",
  "destId": 651,
  "currencyCode": "EUR",
  "searchTypes": ["PRODUCT", "DESTINATION"],
  "text": "sonoma",
  "sortOrder": "PRICE_FROM_A"
}

```

#### topX

For returning results in particular positions. e.g. a value of 11-15 will return the results in position 11 to 15 based on the sort (total of 5 products). Useful for pagination.

#### searchTypes

Specify the type of results:

PRODUCT - a tour / activity

DESTINATION - continent, country, city, region

ATTRACTION - an attraction within a destination (only available to partners with SEO

access)

RECOMMENDATION - recommendation of an attraction from the customer (only available to partners with SEO access)

#### text

The keyword to search on

#### sortOrder

The following sortOrders are available:

TOP\_SELLERS - The top sellers

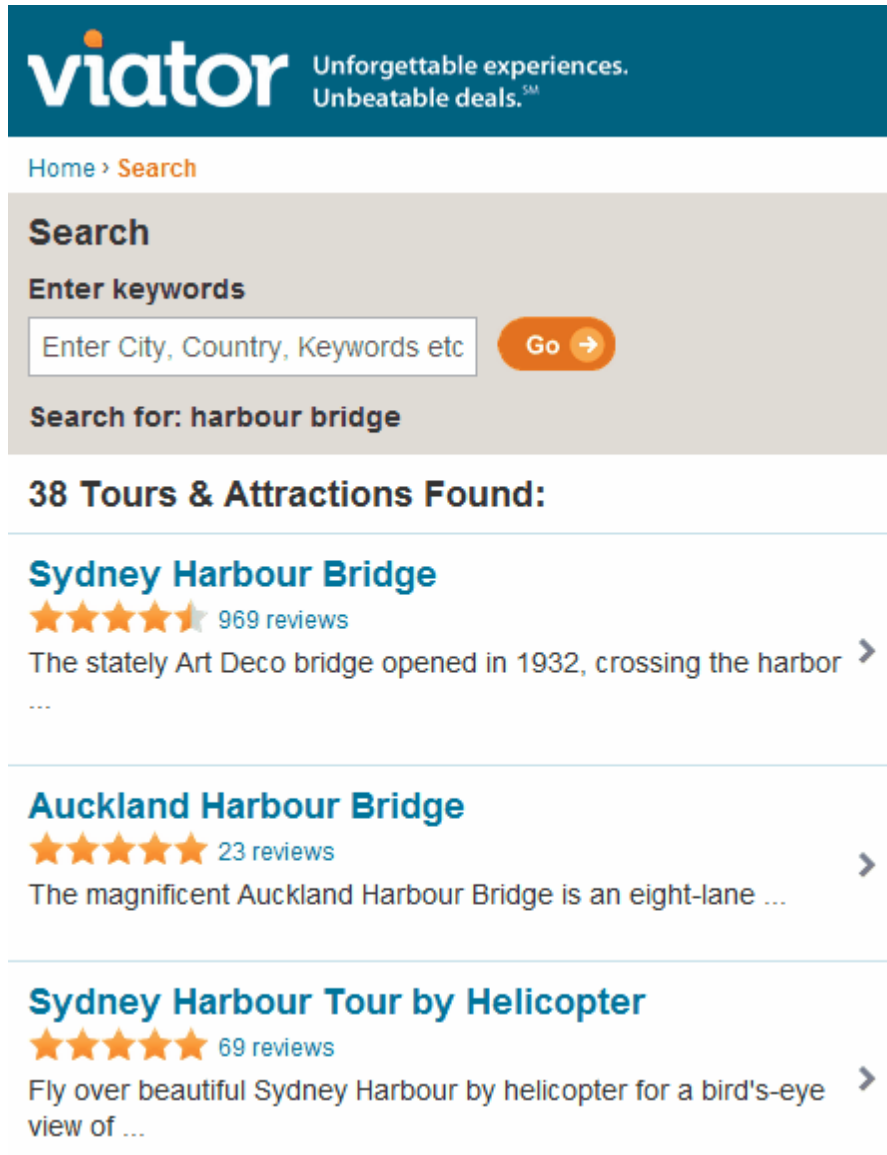
REVIEW\_AVG\_RATING\_A - Traveler Rating (low→high) Average

REVIEW\_AVG\_RATING\_D - Traveler Rating (high→low) Average

PRICE\_FROM\_A - Price (low→high)



Results include a type indicator so you can present appropriately based on the content type.



### Example Result Object

Note the different search types as you can get blended results.

```
{
  "errorReference": null,
  "data": [
    {
      "sortOrder": 1,
      "data": {
        "sortOrder": 1,
        "supplierName": null,
        "currencyCode": "USD",
        "catIds": [
          15
        ],
        "subCatIds": [
          69
        ]
      }
    }
  ]
}
```



```

    ],
    "webURL": "http://prelive.partner.viator.com/en/1234/tours/San-Francisco/San-Fr
    "specialReservationDetails": null,
    "panoramaCount": 0,
    "merchantCancellable": false,
    "bookingEngineId": "UnconditionalBE",
    "onRequestPeriod": null,
    "primaryGroupId": null,
    "shortDescription": "Hop aboard the San Francisco ferry for a trip to Angel Isl
    "price": 8.5,
    "rrp": 0,
    "specialReservation": false,
    "rating": 4.5,
    "thumbnailURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs
    "photoCount": 11,
    "reviewCount": 6,
    "supplierCode": "5863",
    "rrpformatted": "",
    "onSale": false,
    "savingAmount": 0,
    "savingAmountFormatted": "",
    "translationLevel": 0,
    "primaryDestinationId": 651,
    "primaryDestinationName": "San Francisco",
    "priceFormatted": "$8.50",
    "merchantNetPriceFrom": 0,
    "merchantNetPriceFromFormatted": "",
    "specialOfferAvailable": false,
    "thumbnailHiResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/t
    "shortTitle": "San Francisco Ferry: Angel Island",
    "videoCount": 0,
    "title": "San Francisco Ferry: Angel Island",
    "duration": "40 minutes departing from San Francisco<br>55 - 70 minutes departi
    "code": "5863FERANGEL"
  },
  "searchType": "PRODUCT"
},
{
  "sortOrder": 2,
  "data": {
    "sortOrder": 2,
    "supplierName": null,
    "currencyCode": "USD",
    "catIds": [
      6,
      5
    ],
    "subCatIds": [
      21,
      15
    ],
    "webURL": "http://prelive.partner.viator.com/en/1234/tours/San-Francisco/Napa-a
    "specialReservationDetails": null,
    "panoramaCount": 0,
    "merchantCancellable": false,
    "bookingEngineId": "FreesaleBE",
    "onRequestPeriod": null,
    "primaryGroupId": null,
    "shortDescription": "Wine taste your way through Napa and Sonoma on this day tr
    "price": 75.95,
    "rrp": 0,
    "specialReservation": false,
    "rating": 4.5,
    "thumbnailURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs
    "photoCount": 14,
    "reviewCount": 1101,

```



```

        "supplierCode": "2660",
        "rrpformatted": "",
        "onSale": false,
        "savingAmount": 0,
        "savingAmountFormatted": "",
        "translationLevel": 0,
        "primaryDestinationId": 651,
        "primaryDestinationName": "San Francisco",
        "priceFormatted": "$75.95",
        "merchantNetPriceFrom": 0,
        "merchantNetPriceFromFormatted": "",
        "specialOfferAvailable": false,
        "thumbnailHiResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/t",
        "shortTitle": "Napa and Sonoma Wine Country Tour",
        "videoCount": 0,
        "title": "Napa and Sonoma Wine Country Tour",
        "duration": "8 hours",
        "code": "2660SFOWIN"
    },
    "searchType": "PRODUCT"
},
{
    "sortOrder": 3,
    "data": {
        "sortOrder": 3,
        "supplierName": null,
        "currencyCode": "USD",
        "catIds": [
            6
        ],
        "subCatIds": [
            21
        ],
        "webURL": "http://prelive.partner.viator.com/en/1234/tours/San-Francisco/Urban-",
        "specialReservationDetails": null,
        "panoramaCount": 0,
        "merchantCancellable": false,
        "bookingEngineId": "FreesaleBE",
        "onRequestPeriod": null,
        "primaryGroupId": null,
        "shortDescription": "Skip the drive to Napa or Sonoma and instead try tasting w",
        "price": 79,
        "rrp": 0,
        "specialReservation": false,
        "rating": 0,
        "thumbnailURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs",
        "photoCount": 0,
        "reviewCount": 0,
        "supplierCode": "6393",
        "rrpformatted": "",
        "onSale": false,
        "savingAmount": 0,
        "savingAmountFormatted": "",
        "translationLevel": 0,
        "primaryDestinationId": 651,
        "primaryDestinationName": "San Francisco",
        "priceFormatted": "$79.00",
        "merchantNetPriceFrom": 0,
        "merchantNetPriceFromFormatted": "",
        "specialOfferAvailable": false,
        "thumbnailHiResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/t",
        "shortTitle": "Urban Wineries Tour of San Francisco",
        "videoCount": 0,
        "title": "Urban Wineries Tour of San Francisco",
        "duration": "3 hours 30 minutes",
        "code": "6393WINE"
    }
}

```



```

    },
    "searchType": "PRODUCT"
  },
  {
    "sortOrder": 4,
    "data": {
      "sortOrder": 4,
      "supplierName": null,
      "currencyCode": "USD",
      "catIds": [
        12,
        6,
        5
      ],
      "subCatIds": [
        5211,
        21,
        15
      ],
      "webURL": "http://prelive.partner.viator.com/en/1234/tours/San-Francisco/San-Fr
      "specialReservationDetails": null,
      "panoramaCount": 0,
      "merchantCancellable": false,
      "bookingEngineId": "FreesaleOnRequestBE",
      "onRequestPeriod": 24,
      "primaryGroupId": null,
      "shortDescription": "Discover why beautiful Muir Woods and the Californian wine
      "price": 91.8,
      "rrp": 0,
      "specialReservation": false,
      "rating": 4.5,
      "thumbnailURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs
      "photoCount": 103,
      "reviewCount": 678,
      "supplierCode": "2402",
      "rrpformatted": "",
      "onSale": false,
      "savingAmount": 0,
      "savingAmountFormatted": "",
      "translationLevel": 0,
      "primaryDestinationId": 651,
      "primaryDestinationName": "San Francisco",
      "priceFormatted": "$91.80",
      "merchantNetPriceFrom": 0,
      "merchantNetPriceFromFormatted": "",
      "specialOfferAvailable": true,
      "thumbnailHiResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/t
      "shortTitle": "San Francisco Super Saver: Muir Woods and Wine Country Tour",
      "videoCount": 0,
      "title": "San Francisco Super Saver: Muir Woods and Wine Country Tour",
      "duration": "9 hours",
      "code": "2402MUIRWINE"
    },
    "searchType": "PRODUCT"
  },
  {
    "sortOrder": 5,
    "data": {
      "sortOrder": 5,
      "supplierName": null,
      "currencyCode": "USD",
      "catIds": [
        5
      ],
      "subCatIds": [
        15
      ],

```



```

    ],
    "webURL": "http://prelive.partner.viator.com/en/1234/tours/San-Francisco/Wine-C",
    "specialReservationDetails": null,
    "panoramaCount": 0,
    "merchantCancellable": false,
    "bookingEngineId": "FreesaleBE",
    "onRequestPeriod": null,
    "primaryGroupId": null,
    "shortDescription": "Redwoods, hiking, and world-class wine and cheese. Experie",
    "price": 98.1,
    "rrp": 108.99,
    "specialReservation": false,
    "rating": 0,
    "thumbnailURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/thumbs",
    "photoCount": 0,
    "reviewCount": 0,
    "supplierCode": "3454",
    "rrpformatted": "$108.99",
    "onSale": true,
    "savingAmount": 10.89,
    "savingAmountFormatted": "$10.89",
    "translationLevel": 0,
    "primaryDestinationId": 651,
    "primaryDestinationName": "San Francisco",
    "priceFormatted": "$98.10",
    "merchantNetPriceFrom": 0,
    "merchantNetPriceFromFormatted": "",
    "specialOfferAvailable": true,
    "thumbnailHiResURL": "http://cache.graphicslib.viator.com/graphicslib-prelive/t",
    "shortTitle": "Wine Country and Muir Woods Small-Group Tour from San Francisco",
    "videoCount": 0,
    "title": "Wine Country and Muir Woods Small-Group Tour from San Francisco",
    "duration": "10 hours",
    "code": "3454_B0020"
  },
  "searchType": "PRODUCT"
}
},
"dateStamp": "2014-12-03T20:15:41+0000",
"errorType": null,
"errorMessage": null,
"errorName": null,
"success": true,
"totalCount": 29,
"errorMessageText": null,
"vmid": "321001"
}

```

## 5.5 Booking APIs

Making a booking on the Viator environment is somewhat complex due to the variety of tours and other products we sell.

There are a number of key concepts that must be understood in order to develop a booking workflow and application.

### Booking Types - On Request and Freesale

Bookings made with Viator can be either Freesale (immediate confirmation) or On Request where we have to confirm with our product vendor that there is still availability. This must be clearly communicated with the customer during the price check and on the order summary post purchase page.





For Freesale bookings, the voucher is available immediately and their credit card will be charged. For On Request bookings, the confirmation will be sent to the customer within a period that is supplied in the price check and booking API call responses. When the on request booking is confirmed, the customer's credit card will then be charged.

### **Tour Grades**

Tours can have one or more tour grades. Each tour grade might represent a departure time or different tour options such as additional meals, transport or similar. If the tour grade code is 'DEFAULT' do not display this to the consumer, simply hide the tour grade information.

### **Language Options**

Many tours provide delivery in multiple languages via multilingual guides, pre recorded or written guides. Where available, the consumer can pre select the language option.

### **Age Bands**

Most products have multiple age bands such as Adults, Children, Infants, etc. These age bands are returned with the product details API. The consumer can select different quantities of each ageband during price checks and checkout.

### **Passenger Mix**

Some tour grades have defined passenger mixes for pricing of family passes or special mixes for limited passenger quantities such as small buggies or weddings. These passenger mixes are provided in the booking/availability/tourgrades API call. You may need to display these so consumers can understand why they can or can't select various tourgrades if there is a passenger mix mismatch.

### **Pickup Location and Hotel Lists**

Some products have pickup and return shuttles. For these tours, you will need the customer to supply a pickup hotel or else select live locally or hotel not yet booked options.

Viator maintain pickup hotel lists for many popular destinations. These lists are available for consumers to select their pickup location for various tours. For destinations without hotel lists, consumers can enter the name of their hotel. If a customer's hotel is not listed, they should be able to enter a hotel name although pickup may not be possible for that hotel.

### **Lead Traveler**

Each tour booking requires a Lead Traveler to be identified. This is set by the leadTraveller boolean on the traveler class.

### **Booking Questions**

Some products have a list of one or more booking questions that need to be asked. Some are mandatory. The question, a description, etc are provided in the product details object.

The answers need to be included with the booking request.

### **SSL/HTTPS**

Calls to the book API must use a secure channel as they contain credit card information.

### **PromoCodes**



Viator can create promotion codes for discounts or other purposes. It's unlikely that this would be supported for Viator Partners, so just supply NULL and do not include any customer entry fields during the checkout.

### Partner Data

Partners can supply some additional information for their internal purposes. These are attached to booking reports, etc and can be used to allocate commissions to agents.

## 5.5.1 Booking Data Classes

Here's the classes used for booking expressed in PHP syntax:

```
class booking
{
    var $booker;                // see class definition below
    var $ccPayDetail;           // see class definition below
    var $items = array();       // see class definition below
    var $sessionId;
    var $currencyCode;
    var $promoCode;
    var $partnerDetail;        // for partner bookings - this can be null if not used.
    var $newsletterSignUp;
    var $demo;
}

// Partner use only.  For tracking commissions, etc.
class partnerDetail
{
    var $subPUIID;             // is some number which the partner should know
    var $agentNumber;          // is the agent PCC (??) / ARC / IATA number
    var $agentEmailDelivery;   // is one of these options (based off VAP and it seems
    var $additionalEmail;      // is some additional agent email
    var $partnerParam;         // is some extra parameters they want to write to the

}

// Viator use only
class partnerDetailEAP
{
    var $eapChainCode;
    var $eapPropertyCode;
    var $eapAuid;
}

// Viator use only
class partnerDetailShareASale
{
    var $shareSaleId;
    var $shareSaleData;
}

class priceCheck
{
    var $promoCode;
    var $currencyCode;
    var $items = array();
}

class booker
```



```
{
    var $email;
    var $workPhone;
    var $postcode;
    var $homePhone;
    var $firstname;
    var $surname;
    var $title;
}

class ccPayDetail
{
    var $ccaddress1;
    var $ccaddress2;
    var $ccaddressCity;
    var $ccaddressState;
    var $ccaddressZip;
    var $ccphone;
    var $ccctype;
    var $ccname;
    var $ccnumber;
    var $ccaddressCountryId;
    var $ccexpire;
    var $ccadditionalDigits;
}

class item
{
    var $travelDate;
    var $productCode;
    var $tourGradeCode;
    var $hotelId;
    var $pickupPoint;
    var $languageOptionCode;

    var $travellers = array();
    var $bookingQuestionAnswers = array(); // array of bookingQuestionAnswer
    var $specialRequirements;
}

class bookingQuestionAnswer
{
    var $questionId;
    var $answer;
}

class traveller
{
    var $bandId;
    var $firstname;
    var $surname;
    var $title;
    var $leadTraveller;

    var $cellPhoneCountryCode;
    var $cellPhone;
}
```

### 5.5.2 /booking/availability/dates

Method: GET



All available dates for a product excluding days it does not operate and blockouts. This returns a multi dimensional array of year-month -> days that have availability of any tour grade or passenger mix.

### Sample query string

productCode=2280AAHT

This is a very useful call when you want to present a user with a list of dates for selected that include only days the product is available for booking. Note that it might not be available for the passenger mix the user selects, but you can show these details when you get a tourgrade list.

## Leeds Castle Private Viewing, Canterbury and Greenwich Day Trip from London

Product Code: 2452T7

### Pricing & Availability

Choose your travel date and number of passengers to check availability.

#### Select a date:

Note: Only dates that can be booked are shown.

16 ▾ Apr 2011 ▾

#### Enter total number of travelers:

Adult (Ages 16-59) 3 ▾

Senior (Ages 60-99) 0 ▾

Child (Ages 3-15) 0 ▾

Infant (Ages 0-2) 0 ▾

[← Back](#)

[Next →](#)

### Example Result Object

Note that dates have been removed to save trees. Normally there would be many months of days.

```
{
  data: {
    2011-04: [
      "14"
      "15"
      "16"
      "26"
      "27"
      "28"
```



```

        "29"
        "30"
    ]
    2011-05: [
        "01"
        "02"
        "03"
        "04"
        "05"
        "06"
        "27"
        "28"
        "29"
        "30"
        "31"
    ]
}
errorMessage: null
dateStamp: "2011-04-12T09:20:45+0000"
errorReference: null
success: true
errorType: null
totalCount: 1
}

```

### 5.5.3 /booking/availability

Method: POST

This API returns the tour grade with the lowest price that is available for each day.

This can be used when displaying a calendar of available tours. The retailPrice would represent the "from" price for the tour.

#### Example JSON Data

```

{
  "productCode": "2280AAHT",
  "month": "12",
  "year": "2013",
  "currencyCode": "EUR",
  "ageBands": [{"bandId": 1, "count": 1}]
}

```

note: /booking/availability/dates provides all availability in one call and is more suitable for calendars, etc.

#### Example Result Object

```

{
  "data": {
    "productCode": "2280AAHT",
    "availability": [
      {
        "available": true,
        "currencyCode": "USD",
        "retailPrice": 299.99,
        "bookingDate": "2013-12-01",
        "gradeCode": "EARLYM",
        "unavailableReason": null,
        "retailPriceFormatted": "$299.99",
        "merchantNetPrice": 0,
        "merchantNetPriceFormatted": "",
        "sortOrder": 1
      }
    ]
  }
}

```

viator

```

    },
    {
      "available": true,
      "currencyCode": "USD",
      "retailPrice": 329.99,
      "bookingDate": "2013-12-02",
      "gradeCode": "EARLYM",
      "unavailableReason": null,
      "retailPriceFormatted": "$329.99",
      "merchantNetPrice": 0,
      "merchantNetPriceFormatted": "",
      "sortOrder": 2
    },
    {
      "available": true,
      "currencyCode": "USD",
      "retailPrice": 329.99,
      "bookingDate": "2013-12-03",
      "gradeCode": "EARLYM",
      "unavailableReason": null,
      "retailPriceFormatted": "$329.99",
      "merchantNetPrice": 0,
      "merchantNetPriceFormatted": "",
      "sortOrder": 3
    },
    {
      "available": true,
      "currencyCode": "USD",
      "retailPrice": 329.99,
      "bookingDate": "2013-12-04",
      "gradeCode": "EARLYM",
      "unavailableReason": null,
      "retailPriceFormatted": "$329.99",
      "merchantNetPrice": 0,
      "merchantNetPriceFormatted": "",
      "sortOrder": 4
    }
  ],
  "lastAvailableDate": "2014-12-31",
  "firstAvailableDate": "2013-12-01"
},
"vmid": "221001",
"errorMessage": null,
"errorType": null,
"dateStamp": "2013-06-04T16:58:05+0000",
"totalCount": 1,
"errorReference": null,
"errorMessageText": null,
"success": true,
"errorName": null
}

```

#### 5.5.4 /booking/availability/tourgrades

Method: POST

List all available tour grades for a specific day

##### Example JSON Data

```

{
  "productCode": "2280AAHT",
  "bookingDate": "2012-09-04",
  "currencyCode": "EUR",

```



```

    "ageBands": [{"bandId": 1, "count": 3}]
  }

```

[Home](#) › [USA](#) › [Las Vegas](#) › [Tour](#) › [Options](#)

## Grand Canyon All American Helicopter Tour

**Travel Date:** April 16, 2011

**Number of Travelers:** 2 (2 Adults)

**Product Code:** 2280AAHT

### Pricing & Availability

Select your preferred option below

#### Early Morning Departure

Code: EARLYM

Flight departs Las Vegas  
between 7am & 8:00am

[Language Options](#)

Total USD  
**\$640.00**

Price for All Travelers

Select Option →

#### Late Afternoon Departure

Code: LATEA

Flight departs Las Vegas  
between 3:15pm & 4:15pm.  
Available April to August only

[Language Options](#)

Total USD  
**\$593.98**

Price for All Travelers

Select Option →

[← Back](#)

### Common Example for multiple departures in a day

Some interesting features include the multiple departures (each represented as a your grade) and the language options (langServices). This request is for 3 adults on a helicopter tour.

Request object.

```

{
  "productCode": "2280AAHT",
  "bookingDate": "2013-05-11",
  "currencyCode": "EUR",
  "ageBands": [
    {
      "bandId": 1,
      "count": 3
    }
  ]
}

```

Response object.

```

{
  "data": [

```



```

{
  "available": false,
  "ageBands": null,
  "langServices": null,
  "gradeCode": "EARLYM",
  "unavailableReason": "BOOKING_CUTOFF_EXPIRED",
  "gradeTitle": "Early Morning Departure",
  "gradeDepartureTime": "",
  "gradeDescription": "Flight departs Las Vegas between 7am & 8am",
  "defaultLanguageCode": "en",
  "ageBandsRequired": null,
  "currencyCode": "ERROR",
  "retailPrice": 0,
  "bookingDate": "2013-05-11",
  "retailPriceFormatted": "",
  "merchantNetPrice": 0,
  "merchantNetPriceFormatted": "",
  "sortOrder": 1
},
{
  "available": false,
  "ageBands": null,
  "langServices": null,
  "gradeCode": "LATEM",
  "unavailableReason": "BOOKING_CUTOFF_EXPIRED",
  "gradeTitle": "Late Morning Departure",
  "gradeDepartureTime": "",
  "gradeDescription": "Flight departs Las Vegas between 9:45am & 10:45am",
  "defaultLanguageCode": "en",
  "ageBandsRequired": null,
  "currencyCode": "ERROR",
  "retailPrice": 0,
  "bookingDate": "2013-05-11",
  "retailPriceFormatted": "",
  "merchantNetPrice": 0,
  "merchantNetPriceFormatted": "",
  "sortOrder": 2
},
{
  "available": false,
  "ageBands": null,
  "langServices": null,
  "gradeCode": "EARLYA",
  "unavailableReason": "BOOKING_CUTOFF_EXPIRED",
  "gradeTitle": "Early Afternoon Departure",
  "gradeDepartureTime": "2:50 PM",
  "gradeDescription": "Flight departs Las Vegas between 12:30pm & 1:30pm",
  "defaultLanguageCode": "en",
  "ageBandsRequired": null,
  "currencyCode": "ERROR",
  "retailPrice": 0,
  "bookingDate": "2013-05-11",
  "retailPriceFormatted": "",
  "merchantNetPrice": 0,
  "merchantNetPriceFormatted": "",
  "sortOrder": 3
},
{
  "available": false,
  "ageBands": null,
  "langServices": null,
  "gradeCode": "LATEA",
  "unavailableReason": "BOOKING_CUTOFF_EXPIRED",
  "gradeTitle": "Late Afternoon Departure",
  "gradeDepartureTime": "",
  "gradeDescription": "Flight departs Las Vegas between 3:15pm & 4:15pm; available Ap

```





```

        "defaultLanguageCode": "en",
        "ageBandsRequired": null,
        "currencyCode": "ERROR",
        "retailPrice": 0,
        "bookingDate": "2013-05-11",
        "retailPriceFormatted": "",
        "merchantNetPrice": 0,
        "merchantNetPriceFormatted": "",
        "sortOrder": 4
    }
],
"vmid": "221001",
"errorMessage": null,
"errorType": null,
"dateStamp": "2013-06-04T17:01:34+0000",
"totalCount": 1,
"errorReference": null,
"errorMessageText": null,
"success": true,
"errorName": null
}

```

### Example with Passenger Mix Mismatch

The request is for 5 adults. This product only support 1-4 adults.

```

{
  "productCode": "2280ULTWED",
  "bookingDate": "2013-12-11",
  "currencyCode": "EUR",
  "ageBands": [
    {
      "bandId": 1,
      "count": 1
    }
  ]
}

```

So the response has TRAVELLER\_MISMATCH and you can see the ageBandsRequired values for the adult (1) ageband in the available tour grade.

```

{
  "data": [
    {
      "available": false,
      "ageBands": null,
      "langServices": null,
      "gradeCode": "DEFAULT",
      "unavailableReason": "TRAVELLER_MISMATCH",
      "gradeTitle": "DEFAULT",
      "gradeDepartureTime": "12:00 AM",
      "gradeDescription": "DEFAULT",
      "defaultLanguageCode": "en",
      "ageBandsRequired": [
        {
          "bandId": 1,
          "minimumCountRequired": 2,
          "maximumCountRequired": 2
        }
      ],
    }
  ]
}

```



```

        {
            "bandId": 1,
            "minimumCountRequired": 3,
            "maximumCountRequired": 3
        },
        [
            {
                "bandId": 1,
                "minimumCountRequired": 4,
                "maximumCountRequired": 4
            }
        ],
        "currencyCode": "ERROR",
        "retailPrice": 0,
        "bookingDate": "2013-12-11",
        "retailPriceFormatted": "",
        "merchantNetPrice": 0,
        "merchantNetPriceFormatted": "",
        "sortOrder": 1
    }
},
"vmid": "221001",
"errorMessage": null,
"errorType": null,
"dateStamp": "2013-06-04T17:02:35+0000",
"totalCount": 1,
"errorReference": null,
"errorMessageText": null,
"success": true,
"errorName": null
}

```

### 5.5.5 /booking/availability/tourgrades/pricingmatrix

Method: POST

Availability, Tour Grade Availabilty, Pricing Matrix Hybrid

Given a month this service will return available days only (ie days which have at least one tourgrade available), and the pricing matrix for that tour grade for that day.

#### Example JSON Data

```

{
    "productCode": "2280AAHT",
    "month": "05", "year": "2013",
    "currencyCode": "EUR"
}

```

#### Example Result Object

```

{
    "data": {
        "bookingMonth": "2013-05",
        "pricingUnit": "per person",
        "dates": [
            {
                "tourGrades": [
                    {
                        "pricingMatrix": [
                            {
                                "sortOrder": 1,

```



```

        "bookingDate": "2013-05-30",
        "pricingUnit": "per person",
        "ageBandPrices": [
            {
                "sortOrder": 1,
                "prices": [
                    {
                        "currencyCode": "EUR",
                        "sortOrder": 1,
                        "price": 257.63,
                        "priceFormatted": "€257,63",
                        "minNoOfTravellersRequiredForPrice": 1
                    }
                ],
                "bandId": 1,
                "minimumCountRequired": 1,
                "maximumCountRequired": null
            },
            {
                "sortOrder": 2,
                "prices": [
                    {
                        "currencyCode": "EUR",
                        "sortOrder": 1,
                        "price": 0,
                        "priceFormatted": "€0,00",
                        "merchantNetPrice": 0,
                        "merchantNetPriceFormatted": "€0,00",
                        "minNoOfTravellersRequiredForPrice": 1
                    }
                ],
                "bandId": 2,
                "minimumCountRequired": 0,
                "maximumCountRequired": null
            }
        ]
    },
    "gradeCode": "EARLYM",
    "gradeTitle": "Early Morning Departure",
    "sortOrder": 1
},
{
    "pricingMatrix": [
        {
            "sortOrder": 1,
            "bookingDate": "2013-05-30",
            "pricingUnit": "per person",
            "ageBandPrices": [
                {
                    "sortOrder": 1,
                    "prices": [
                        {
                            "currencyCode": "EUR",
                            "sortOrder": 1,
                            "price": 257.63,
                            "priceFormatted": "€257,63",
                            "merchantNetPrice": 0,
                            "merchantNetPriceFormatted": "€0,00",
                            "minNoOfTravellersRequiredForPrice": 1
                        }
                    ],
                    "bandId": 1,
                    "minimumCountRequired": 1,
                    "maximumCountRequired": null
                }
            ]
        }
    ]
}

```

```

    },
    {
        "sortOrder": 2,
        "prices": [
            {
                "currencyCode": "EUR",
                "sortOrder": 1,
                "price": 0,
                "priceFormatted": "€0,00",
                "merchantNetPrice": 0,
                "merchantNetPriceFormatted": "€0,00",
                "minNoOfTravellersRequiredForPrice": 1
            }
        ],
        "bandId": 2,
        "minimumCountRequired": 0,
        "maximumCountRequired": null
    }
]
},
{
    "gradeCode": "LATEM",
    "gradeTitle": "Late Morning Departure",
    "sortOrder": 2
},
{
    "pricingMatrix": [
        {
            "sortOrder": 1,
            "bookingDate": "2013-05-30",
            "pricingUnit": "per person",
            "ageBandPrices": [
                {
                    "sortOrder": 1,
                    "prices": [
                        {
                            "currencyCode": "EUR",
                            "sortOrder": 1,
                            "price": 257.63,
                            "priceFormatted": "€257,63",
                            "merchantNetPrice": 0,
                            "merchantNetPriceFormatted": "€0,00",
                            "minNoOfTravellersRequiredForPrice": 1
                        }
                    ],
                    "bandId": 1,
                    "minimumCountRequired": 1,
                    "maximumCountRequired": null
                },
                {
                    "sortOrder": 2,
                    "prices": [
                        {
                            "currencyCode": "EUR",
                            "sortOrder": 1,
                            "price": 0,
                            "priceFormatted": "€0,00",
                            "merchantNetPrice": 0,
                            "merchantNetPriceFormatted": "€0,00",
                            "minNoOfTravellersRequiredForPrice": 1
                        }
                    ],
                    "bandId": 2,
                    "minimumCountRequired": 0,
                    "maximumCountRequired": null
                }
            ]
        }
    ]
}

```



```

        ]
    },
    "gradeCode": "EARLYA",
    "gradeTitle": "Early Afternoon Departure",
    "sortOrder": 3
},
{
    "pricingMatrix": [
        {
            "sortOrder": 1,
            "bookingDate": "2013-05-30",
            "pricingUnit": "per person",
            "ageBandPrices": [
                {
                    "sortOrder": 1,
                    "prices": [
                        {
                            "currencyCode": "EUR",
                            "sortOrder": 1,
                            "price": 257.63,
                            "priceFormatted": "€257,63",
                            "merchantNetPrice": 0,
                            "merchantNetPriceFormatted": "€0,00",
                            "minNoOfTravellersRequiredForPrice": 1
                        }
                    ],
                    "bandId": 1,
                    "minimumCountRequired": 1,
                    "maximumCountRequired": null
                },
                {
                    "sortOrder": 2,
                    "prices": [
                        {
                            "currencyCode": "EUR",
                            "sortOrder": 1,
                            "price": 0,
                            "priceFormatted": "€0,00",
                            "merchantNetPrice": 0,
                            "merchantNetPriceFormatted": "€0,00",
                            "minNoOfTravellersRequiredForPrice": 1
                        }
                    ],
                    "bandId": 2,
                    "minimumCountRequired": 0,
                    "maximumCountRequired": null
                }
            ]
        }
    ],
    "gradeCode": "LATEA",
    "gradeTitle": "Late Afternoon Departure",
    "sortOrder": 4
}
],
"bookingDate": "2013-05-30",
"callForLastMinAvailability": true,
"sortOrder": 2
}
],
},
"vmid": "221001",
"errorMessage": null,
"errorType": null,
"dateStamp": "2013-05-27T17:49:53+0000",

```



```

    "totalCount": 1,
    "errorReference": null,
    "errorMessageText": null,
    "success": true,
    "errorName": null
  }

```

### 5.5.6 /booking/calculateprice

Method: POST

This API is used to calculate a total price taking into account any promo codes (viator only) and passenger mix. It also reconfirms the availability of the products for the dates and passenger mix provided as well as pricing. It is strongly recommended that this service is called prior to making the booking to confirm that the booking will go through. This can be used for a shopping cart.

#### Example JSON Data

```

{
  "currencyCode": "USD",
  "items": [
    {
      "travelDate": "2015-03-01",
      "productCode": "2916ROME",
      "tourGradeCode": "24HR",
      "travellers": [
        {
          "bandId": 1
        },
        {
          "bandId": 1
        }
      ]
    }
  ]
}

```

The travelers are an associative array of bandId -> qty

There can be several items in the cart. This allows for different passenger mixes per product booked.

The overall total of the booking is displayed in the itineraryNewPrice and itineraryNewPriceFormatted fields.

The result indicates if the booking is Freesale or On Request. On Request bookings are not sold as confirmed but are confirmed by the supplier at a later date. The approximate window for confirmation is provided in the hoursConfirmed (integer) field for presentation to the customer. An hoursConfirmed of 0 means that the booking is Freesale. An hoursConfirmed greater than 0 indicates that it is On Request.

The partnerDetail object can also be added to the JSON. Please see /booking/book for more details on the partnerDetails object.

Freesale example:



## Leeds Castle Private Viewing, Canterbury and Greenwich Day Trip from London

**Travel Date:** April 16, 2011

**Number of Travelers:** 3 (2 Adults, 1 Senior)

**Product Code:** 2452T7

### Pricing & Availability

Review Your Order & Proceed To Checkout

**Your Current Total:**

USD  
**\$349.12**

Enter promo code and tap 'Apply Code':

**Apply Code**

*If you don't have a promo code, leave blank.*

This activity will be charged to the credit card upon completion of booking.

 [Back](#)

[Proceed to Checkout](#) 

On Request Example - note the different way they are presented to the user. The approximate number of hours are provided in the API result.



[Home](#) › [USA](#) › [Las Vegas](#) › [Tour](#) › [Review Order](#)

## Grand Canyon All American Helicopter Tour

**Travel Date:** April 16, 2011

**Number of Travelers:** 2 (2 Adults)

**Selected Option:** Early Morning Departure

**Product Code:** 2280AAHT

### Pricing & Availability

Review Your Order & Proceed To Checkout

**Your Current Total:**

USD  
**\$640.00**

Enter promo code and tap 'Apply Code':

**Apply Code**

*If you don't have a promo code, leave blank.*

Confirmation of this activity will take up to 48 hours, depending on availability.

[← Back](#)

[Proceed to Checkout →](#)

### Response Object

```
{
  "errorReference": null,
  "data": {
    "rulesApplied": [],
    "promoCode": null,
    "itinerary": {
      "sortOrder": 0,
      "rulesApplied": [],
      "bookingStatus": {
        "status": 0,
        "text": "Waiting to be Booked",
        "type": "WAITING",
        "level": "ITINERARY",
        "confirmed": false,
        "pending": false,
        "amended": false,
        "cancelled": false,
        "failed": false
      },
      "itemSummaries": [{
        "sortOrder": 0,
        "rulesApplied": [],
        "bookingStatus": {
          "status": 0,
          "text": "Waiting to be Booked",
          "type": "WAITING",
          "level": "ITEM",
          "confirmed": false,
```





```
        "failed": false,
        "confirmed": false,
        "amended": false,
        "cancelled": false
    },
    "travellerAgeBands": [{
        "sortOrder": 0,
        "count": 1,
        "pluralDescription": "Adults",
        "description": "Adult",
        "ageBandId": 1
    }],
    "voucherKey": "",
    "voucherURL": null,
    "voucherRequirements": "You must present a paper voucher for this tour",
    "productPulledDown": false,
    "merchantCancellable": false,
    "productWidgetList": null,
    "savingAmount": 0,
    "passbooks": null,
    "termsAndConditions": null,
    "itineraryId": 1000024809,
    "tourGradeCode": "1DAYBOAT",
    "productCode": "2065CPT",
    "leadTravellerSurname": null,
    "distributorItemRef": null,
    "languageServicesLanguageCode": null,
    "travelDate": "2016-02-01",
    "price": 4.51,
    "bookingEngineId": "UF",
    "merchantNetPrice": 0,
    "merchantNetPriceFormatted": "",
    "leadTravellerTitle": null,
    "leadTravellerFirstname": "",
    "lastRetailPrice": 4.51,
    "destId": 318,
    "voucherOption": "VOUCHER_PAPER_ONLY",
    "productTitle": "Cape Town City Hop-on Hop-off Tour",
    "itemId": 700025516,
    "barcodeOption": "perperson",
    "barcodeType": "code128",
    "obfsId": 9779,
    "priceFormatted": "$4.51",
    "savingAmountFormatted": "$0.00",
    "priceUSD": 4.51,
    "lastRetailPriceFormatted": "$4.51",
    "departsFrom": "Cape Town, South Africa",
    "hoursConfirmed": 0,
    "currencyCode": "USD"
}],
    "voucherURL": null,
    "voucherKey": "",
    "bookerEmail": null,
    "userId": null,
    "itineraryId": 1000024809,
    "exchangeRate": 1,
    "totalPriceFormatted": "$4.51",
    "totalPriceUSD": 4.51,
    "bookingDate": "2015-03-17",
    "distributorRef": null,
    "totalPrice": 4.51,
    "hasVoucher": false,
    "currencyCode": "USD"
},
    "promoCodeValid": false,
    "promoCodeExpired": false,
```



```
        "itineraryFromPriceFormatted": "$4.51",
        "itineraryNewPrice": 4.51,
        "itineraryNewPriceFormatted": "$4.51",
        "itinerarySaving": 0,
        "itinerarySavingFormatted": "$0.00",
        "hasPromoCode": false,
        "itineraryFromPrice": 4.51,
        "currencyCode": "USD"
    },
    "dateStamp": "2015-03-17T18:06:52+0000",
    "errorType": null,
    "errorMessage": null,
    "errorName": null,
    "success": true,
    "totalCount": 1,
    "vmid": "321003",
    "errorMessageText": null
}
```

### 5.5.7 /booking/hotels

Method: GET



List the hotel pickups available for either a product or a destination

#### Sample URL parameters

```
productCode=2280AAHT
or
destId=684
```

Hotel pickup example:



Go 

### Hotel Pickups

Once your booking is confirmed, we'll send you a confirmation voucher with full contact information (phone number, email address, etc.) for our local operator in case you need to update or change your hotel pick-up details.

---

**My hotel is not listed**>

---

**I live locally/staying with friends or relatives**>

---

**My hotel is not yet booked**>

---

A - F

G - K

L - P

Q - S

T - Z

**A - F**

**Alexis Park Resort Hotel**  
375 East Harmon Avenue>

---

**Americas Best Value Inn**  
167 East Tropicana Avenue>

---

### Example Result Object

```
{
  data: [
    {
      address: null,
      name: "I live locally / I'm staying with friends, relatives",
      id: "local",
      phone: null,
      productCodes: null,
      destinationId: 0,
      city: null,
      notes: null,
      latitude: null,
      longitude: null,
      postcode: null,
      brand: null,
      hotelString: "I live locally / I'm staying with friends, relatives",
      sortOrder: 1
    },
    {
      address: null,
      name: "My hotel is not yet booked",
      id: "notBooked",
      phone: null,
      productCodes: null,
      destinationId: 0,
      city: null,
```



```

        notes: null,
        latitude: null,
        longitude: null,
        postcode: null,
        brand: null,
        hotelString: "My hotel is not yet booked",
        sortOrder: 2
    },
    {
        address: null,
        name: "My hotel is not listed",
        id: "notListed",
        phone: null,
        productCodes: null,
        destinationId: 0,
        city: null,
        notes: null,
        latitude: null,
        longitude: null,
        postcode: null,
        brand: null,
        hotelString: "My hotel is not listed",
        sortOrder: 3
    },
    {
        address: "375 East Harmon Avenue",
        name: "Alexis Park Resort Hotel",
        id: "684_2",
        phone: "",
        productCodes: null,
        destinationId: 684,
        city: "Las Vegas",
        notes: null,
        latitude: 36.106258,
        longitude: -115.156146,
        postcode: "89169",
        brand: "",
        hotelString: null,
        sortOrder: 4
    },
    {
        address: "167 East Tropicana Avenue",
        name: "Americas Best Value Inn",
        id: "684_3",
        phone: "",
        productCodes: null,
        destinationId: 684,
        city: "Las Vegas",
        notes: null,
        latitude: 36.100778,
        longitude: -115.165522,
        postcode: "89109",
        brand: "",
        hotelString: null,
        sortOrder: 5
    },
    {
        address: "3131 Las Vegas Boulevard South",
        name: "Wynn Resort",
        id: "684_126",
        phone: "",
        productCodes: null,
        destinationId: 684,
        city: "Las Vegas",
        notes: null,
        latitude: 36.127563,

```



```
        longitude: -115.167704,
        postcode: "89109",
        brand: "",
        hotelString: null,
        sortOrder: 119
    }
},
vmid: "221002",
errorMessage: null,
errorType: null,
dateStamp: "2012-04-12T13:48:27+0000",
success: true,
errorReference: null,
errorMessageText: null,
totalCount: 1,
errorName: null
}
```

### 5.5.8 /booking/book - partners

Method: POST

**Note:** To make a real booking, ensure **demo** is set to **false** in the booking request.

Make a Booking (HTTPS ONLY).

#### Example JSON Data

```
{
  "demo": true,
  "currencyCode": "USD",
  "booker": {
    "email": "apitest@viator.com",
    "homePhone": "(02) 66987564",
    "firstname": "Homer",
    "surname": "Simpson Test",
    "title": "Mr",
    "cellPhoneCountryCode": "0055",
    "cellPhone": "12345678"
  },
  "ccPayDetail": {
    "ccaddress1": "742 Evergreen Terrace",
    "ccaddress2": "",
    "ccaddressCity": "Springfield",
    "ccaddressState": "WI",
    "ccaddressZip": "12345",
    "ccphone": "",
    "cctype": "Visa",
    "ccname": "Homer Simpson Test",
    "ccnumber": "4242424242424242",
    "ccaddressCountryId": "US",
    "ccexpire": "12/2020",
    "ccadditionalDigits": "123",
    "fiscalNumber": null
  },
  "items": [
    {
      "hotelId": null,
      "pickupPoint": null,
      "travelDate": "2015-03-31",
      "productCode": "2916HOME",
      "tourGradeCode": "24HR",

```



```


"languageOptionCode": "en/SERVICE_GUIDE",
"bookingQuestionAnswers": [
  {
    "questionId": 100,
    "answer": "50 kgs"
  }
],
"travellers": [
  {
    "bandId": 1,
    "firstname": "Homer",
    "surname": "Simpson Test",
    "title": "Mr",
    "leadTraveller": true,
    "cellPhoneCountryCode": "0055",
    "cellPhone": "911"
  },
  {
    "bandId": 1,
    "firstname": "Marge",
    "surname": "Simpson",
    "title": "Mrs"
  }
]
}

```

#### Description of JSON request parameters

Object	Parameter	Type	Comments	Mandatory
	sessionId	varchar	Viator API User Account session Id. Not applicable to partners.	N
	ipAddress	varchar	The client ip address which is doing the booking. Internal sites only	N
	aid	varchar	Viator only field	N
	newsletterSign Up	boolean	If this is set to True, the booker email will be added to the list of subscribers for any Viator marketing emails.	Optional
	demo	boolean	If this is set to True, then it is a demo booking only. Full demos do not send any notifications, are automatically confirmed and OnRequest products become freesale products. Default value is true. Production must have demo set to false.	N
	promoCode	varchar	Viator only field	N
	currencyCode	varchar	The currency the booking will be submitted in. You will be billed in this currency.	N



<b>partnerDetail</b>		object	Applicable only for extra partner detail for either partner, EAP or Share-A-Sale. Itinerary level. Can omit or pass null if not needed.  AAA use the below partner details json: { "subPUID" : "", "agentNumber" : "", "agencyName" : "", "agentName" : "", "agentEmail" : "", "agentEmailDelivery" : "", "additionalEmail" : "", "partnerParam" : "" }  ARC use the below partner details json: { "agentNumber" : "", "agencyName" : "", "agentName" : "", "agentEmail" : "", "agentEmailDelivery" : "" }  Other partner parameters: { ... "subPUID" : "", "partnerParam" : "", "additionalEmail" : "", ... }	N
	agentEmailDelivery		applicable to ARC/AAA (agent partners) Use one of these: "Agent only", "Agent and Customer", "Customer only"	Optional
	subPUID	varchar	Used by agents to pass an additional identifier for their bookings. Alphanumeric characters only.	Optional
	agentNumber	varchar	Agent ID	Optional
	agencyName	varchar	Name of agency	Optional
	agentName	varchar	Name of agent	Optional
	agentEmail	varchar	Email of agent	Optional
	additionalEmail	varchar	Additional email to send the booking confirmation email to.	Optional
	partnerParam	varchar	Additional identifier for bookings for all partners that can handle special characters	Optional
<b>partnerItemDetail</b>		object	Not applicable to API partners	N
<b>otherDetail</b>		object	Other optional details to pass to the booking. At the moment, only call center messages	N
	callCenterSource		Viator only field.	N
	callCenterAgent		Viator only field.	N
<b>booker</b>		object	The information of the primary contact. This contact does not have to be a traveller.	Y
	email	varchar	Email address of the primary contact	Y
	homePhone	varchar	Home phone number of the primary contact	Y
	firstname	varchar	First name of the primary contact	Y
	surname	varchar	Surname of the primary contact. Requires a minimum of 2 characters.	Y
	title	varchar	Title of the primary contact	N
	cellPhoneCountryCode	varchar	Country code of the primary contact's cell phone	N
	cellPhoneCountryCode	varchar	Cell phone number of the primary contact	N
<b>ccPayDetail</b>			 API partners must pass this object. The billing details of the booking	Y

	ccaddress1	varchar	Billing address	Y
	ccaddress2	varchar	Billing address, second line	N
	caddressCity	varchar	City in billing address	Y
	ccaddressState	varchar	See Appendix for US, Canada and Australian states	Y
	ccaddressZip	varchar	Zip / postcode of billing address	Y
	ccaddressCountryId	varchar	See Appendix or service /util/countrymap for countryIds	Y
	ccphone	varchar	Billing phone number	N
	cctype	varchar	Only the following cards are accepted - Visa - "visa" Mastercard - "mastercard" Amex - "amex" Discover - "discover"	Y
	ccname	varchar	Name on credit card	Y
	ccnumber	varchar	Credit card number	Y
	ccexpire	varchar	format mm/yyyy e.g. 01/2014	Y
	ccadditionalDigits	varchar	Equivalent to the ccv number	Y
	fiscalNumber	varchar	This field is <b>mandatory</b> for payments made with a <b>Brazilian</b> credit card, or a transaction conducted in Brazil. The transaction will be rejected by the payment gateway if the field is not provided. The customer name and fiscal number combination must be valid. Note that this number is also referred to as Cadastro de Pessoas Físicas (CPF) or individual fiscal registration number. This field accepts a max of 15 characters.	Y - For transactions in Brazil, N for all other transactions.
<b>altContactDetails</b>		object	Optional object for capturing additional contact details of the primary contact whilst travelling <pre>{   "contactType": "EMAIL",   "phone": "1234675535",   "phoneCountry": 61,   "email": "t@test.com" }</pre> If you specify mobile or alternate, pass phone/phone country If you specify email address, pass in the email address	Optional
	contactType	varchar	Type of contact information passed. The types available are: MOBILE - mobile/cell phone number EMAIL - alternate email address ALTERNATE - Alternate phone number NON_CONTACTABLE - Additional details not available	Optional
	phone	varchar	Alternate phone number or mobile/cell phone number. Must be supplied if contactType is MOBILE or ALTERNATE	Optional
	phoneCountry	varchar	Country code of phone number or mobile/cell phone number. Must be supplied if contactType is MOBILE or ALTERNATE	Optional
	email	varchar	Alternate email address. Must be supplied if contactType is EMAIL	Optional



<b>items</b>			Array of items in itinerary to be booked	Y
	productCode	varchar	product code of the item to be booked	Y
	tourGradeCode	varchar	tourGradeCode of the item to be booked. If tour grades are supplied in /product, you must allow the customer to select a tour grade code. If no tour grades are available for the product, pass DEFAULT.	Y
	languageOptionCode	varchar	The language service provided for this product that has been chosen for this booking. Usually in the format langcode/Service eg en/SERVICE_GUIDE. If the product details service /product for the product returns a langService, this must be provided.	Y - if languageService is provided in /product
	travelDate	date	date of travel for the item. Format is yyyy-mm-dd e.g. 2013-12-25	Y
	hotelId	varchar	If /product returns hotelPickup: true and a list of hotels is available for the product in /booking/hotels, a hotelId must be captured. The hotel id as per the hotel service or use one of these alternative hotel id's: local - Live locally notBooked - Not booked the hotel yet notListed - Hotel not listed!	Y - if /product returns hotelPickup: true for product Code and hotels available
	pickupPoint	varchar	Pickup point information related to hotel pickup. Details of the hotel or pickup point must be provided if the hotelId is selected by the user is "notListed" or if no hotels are returned for the product in /booking/hotels where hotelPickup : true	Y - if hotelId = notListed or no hotels
	specialReservation	boolean	Not applicable to partners	N
	specialRequirements	mediumtext	Capture any additional requirements for the booking, such as dietary requirements or if a wheelchair is required. Suggested workflow is if there are no bookingQuestionAnswers for the product, to collect specialRequirements.	Optional
<b>travellers</b>			Array of traveller names with a required lead traveller selected per item.	Y
	bandId	int	Ageband Id. Available ageband details for the product is listed in /product.	Y
	firstName	varchar	First name of the traveller.	Y
	surname	varchar	Surname of the traveller. Requires a minimum of 2 characters	Y
	title	varchar	Title of the traveller. Suggested options: Mr, Mrs, Ms, Miss, Mstr, Dr	Optional
	leadTraveller	boolean	Each item must have one traveller assigned as the lead traveller for the tour. The lead traveller will have a value of true, all other travellers will have a value of false. The lead traveller can have a mobile phone number added to the booking.	Y

	cellPhoneCountryCode	varchar	Ideally only collect the phone number country code for the lead traveller. Alternatively, collect the phone number of the booker instead.	Optional
	cellPhone	varchar	Ideally only collect the phone number for the lead traveller. Alternatively, collect the phone number of the booker instead.	Optional
<b>booking Question Answers</b>		object	Answers to booking questions for the particular item. - If a booking question is available in the bookQuestions field in /product for the product, the matching bookingQuestionAnswers must be passed. - If a product does not have a bookingQuestion, you can omit the bookingQuestionAnswers field completely. - Any invalid or unrequired bookingQuestionAnswers that are passed to /booking/book will be ignored (no exceptions will be raised)	Y - if /product returns bookQuestions
	questionId	int	QuestionId provided in /product	Y
	answer	varchar	Answer to the question at the questionId listed. Recommended length of 500 characters.	Y
<b>marketing</b>		object	Object to send marketing data associated with the booking to the Viator system. Currently only capturing the user's operating system.	Optional
	os	varchar	The user's operating system. The API partner will need to capture the user's operating system or user agent to pass into this field.	Optional

### Example checkout page workflow

#### Traveler Details



## Grand Canyon All American Helicopter Tour

**Travel Date:** April 16, 2011

**Number of Travelers:** 2 (2 Adults)

**Selected Option:** Early Morning Departure

**Product Code:** 2280AAHT

**Your Current Total:**

USD  
**\$640.00**

### Checkout

TRAVELER  
DETAILS

TOUR  
OPTIONS

BILLING  
DETAILS

#### Enter Traveler Details

\* Required Fields

##### Lead Traveler

First Name: \*

David

Last Name: \*

Challoner



Email Address: \*

*Please note: We will send your voucher to this email address. We recommend using an address accessible from your mobile device.*

Home Phone Number: \*

*Please include your full phone number, including country code.*

Mobile Phone Number:

*We recommend including your mobile phone number, especially if you have already left home. Viator's Customer Care team or our local supplier will only use this number to contact you about issues directly relating to your booking.*

### Additional Adult 2

First Name: \*

Last Name: \*

[← Back](#)

[Next →](#)



## Grand Canyon All American Helicopter Tour

**Travel Date:** April 16, 2011

**Number of Travelers:** 2 (2 Adults)

**Selected Option:** Early Morning Departure

**Product Code:** 2280AAHT

**Your Current Total:**

USD  
**\$640.00**

### Checkout

TRAVELER  
DETAILS ✓

TOUR  
OPTIONS

BILLING  
DETAILS

#### Enter Tour Options

\* Required Fields

Tour Language: \*

English - Audio



Passenger Weights \*

*For safety reasons you must enter the weight of **all** passengers. Please indicate pounds or kilos. (e.g. 127 pounds, 145 kilos, etc)*

Hotel Pickup: \*

Select Hotel Pick-up →

← Back

Next →

Hotel Details



## Hotel Pickups

Once your booking is confirmed, we'll send you a confirmation voucher with full contact information (phone number, email address, etc.) for our local operator in case you need to update or change your hotel pick-up details.

[My hotel is not listed](#)[I live locally/staying with friends or relatives](#)[My hotel is not yet booked](#)[A - F](#)[G - K](#)[L - P](#)[Q - S](#)[T - Z](#)

### A - F

[Alexis Park Resort Hotel](#)

375 East Harmon Avenue

[Americas Best Value Inn](#)

167 East Tropicana Avenue

[Aria Resort and Casino](#)

3730 Las Vegas Boulevard South

[Atrium Suites Hotel](#)

4255 Paradise Road



#### Other Hotel

Hotel Pickup: \*

#### My hotel is not listed

Please enter the name and address of the hotel below for the operator's reference. You will need to contact the operator prior to your tour/activity to verify your closest pick-up point and time. We'll send you complete contact details for our local operator as soon as your booking is complete. Alternatively you can select a hotel from the list.

You need to input hotel name

[Change Hotel Pick-up](#)[← Back](#)[Next →](#)

#### Payment Method



## Grand Canyon All American Helicopter Tour

**Travel Date:** April 16, 2011

**Number of Travelers:** 2 (2 Adults)

**Selected Option:** Early Morning Departure

**Product Code:** 2280AAHT

**Your Current Total:**

USD  
**\$640.00**

### Checkout

TRAVELER  
DETAILS ✓

TOUR  
OPTIONS ✓

BILLING  
DETAILS

#### Enter Billing Details

\* Required Fields

Name on Card: \*

David Challoner

Billing Address (Line 1): \*

900 test ave

Billing Address (Line 2):

City: \*

Zurich



Province/State/County: \*

Not Applicable

Postcode/Zip: \*

300ZUR2

Credit Card Number: \*

*We accept MasterCard, Visa, American Express.*

CCV Number: \*

Expiry Date: \*

January



2013



### Voucher Info

You must present a paper voucher for this tour. We will email a link to access and print your voucher at the Lead Traveler's email address.

### Newsletter

☐ Yes! Sign me up to receive Viator email newsletters & deal alerts.

### Terms & Conditions

☒ I agree to Viator's terms and conditions.

[Back](#)

[Book Now](#)

#### 5.5.8.1 Test Bookings for API partners

##### Test bookings

A demo booking can be made by either setting the demo card to true, or passing a demo credit card number.

If demo is set to **true**, regardless of the credit card number used, the booking will be a **full demo booking**. It is advised that a demo card is used if demo is set to true as it is an extra check or confirmation that a demo booking has been made.

Setting demo to **false** and using the partial demo credit card will make a **partial demo booking**. A partial demo booking will honour the booking type (freesale or on request) and no notifications will be sent to the supplier.

If you use the full demo credit card (4242), the booking will be a full demo booking and the value of the demo flag will be ignored.





Demo card type	Credit card number	Credit card type	Expiry date	Demo flag
Full demo card	4242424242424242	VISA	Any future date	True
Partial demo card	4111111111111111	VISA	Any future date	False

In addition, ensure the ccname includes "test" (e.g. John Test) and indicate in the special requirements it is a test booking.

Test bookings can be made on both prelive and live environments, however if a test booking is made on Live, you must put in a request to cancel the booking. This can be done via the /booking/mybookings/message service.

**NOTE:** If demo is set to false and a real credit card is used in the prelive environment, the credit card will be charged. If you need to test a live credit card on prelive and need the charge refunded, please email [apitechsupport@viator.com](mailto:apitechsupport@viator.com), ideally within 24 hours.

### 5.5.8.2 Errors

A number of errors may be returned in the response to the /booking/book service. If any errors are reported, NO items will be booked, NO items/data will be returned in the response, and NO billing will occur.

There are two error types:

VALIDATION - occurs if a required field is missing or a field is not formatted properly

EXCEPTION - occurs when there are issues with the booking date, product / tourgrade code or the payment.

Example of an error message:

```
{
  "data": null,
  "vmid": "221001",
  "errorMessage": [
    "Email is required"
  ],
  "errorType": "VALIDATION",
  "dateStamp": "2013-04-24T15:25:22+0000",
  "errorReference": null,
  "errorMessageText": [
    "Email is required"
  ],
  "success": false,
  "totalCount": 1,
  "errorName": null
}
```

Please see "Standard JSON fields" in the Appendix for an explanation of the fields.

Below are some errors that you may come across:

scenario	errorType	Error message text
Missing required field from booking request, such as email address	VALIDATION	Email is required
Expiry date of credit card not in format mm/yyyy	VALIDATION	The expiration date for your credit card is not formatted properly. Please verify and re-enter the expiration date.

Lead traveller is not specified	VALIDATION	A traveler needs to be selected as lead traveler. Lead Traveler's name must match credit card name.
Traveller names are not provided	VALIDATION	First name of traveler 1 is required, Last name of traveler 1 is required
No travellers provided	VALIDATION	A traveler needs to be selected as lead traveler. Lead Traveler's name must match credit card name.
Invalid email address	VALIDATION	Your email address format is invalid
Product code does not exist	EXCEPTION	We're sorry, we cannot find the tour, activity or attraction you are looking for
Product code exists, but tour grade code does not exist	EXCEPTION	SICInvalidTourGrade
Booking date in the past	EXCEPTION	We're sorry, the following tour you are trying to book is sold out and no longer available: Grand Canyon All American Helicopter Tour (2280AAHT)
Testing (where demo: true) with a test card not on the list	EXCEPTION	Demo bookings only allow demo cards to be used! Please check the CC no entered
languageOptionCode is in the wrong format	EXCEPTION	LanguageOptionCode should be LangCode/LangServices
Missing required answers for item	EXCEPTION	Additional questions missing
Unsupported currency	EXCEPTION	Could not lookup SGD:java.lang.RuntimeException: Could not lookup SGDf:au.com.fim.v3.etravel.PiusRecordNotFoundFoundException: No currency found :select * from CurrencyFormat where currencyID = 'SGD'
Invalid credit card number, expiry date, and/or CVV	EXCEPTION	We're sorry, we cannot process the credit card you submitted. Either payment has been refused, or the card is invalid. Please verify and re-enter the credit card details, or use a different credit card.

### 5.5.9 /booking/status

Method: POST

This API retrieves the booking status for multiple items based on different criteria.

This service can only be polled every **30 minutes**. This would ideally be used in software for bulk updates of pending itineraries.

The maximum number of results returned is 1000 itineraries. Narrow your search if you expect results greater than this.

**NOTE:** This will return both live and test bookings.

Example JSON Data

```
{
  "bookingDateFrom": "2013-03-22",
  "viator"
```

```

    "bookingDateTo": "2013-03-25",
    "itineraryIds": null,
    "itemIds": null,
    "distributorRefs": ["Ref20132603_1", "Ref20132603_5"],
    "distributorItemRefs": null,
    "leadFirstName": null,
    "leadSurname": null,
    "test": true
}

```

All fields are optional and can be omitted, however at least one needs to be provided.

#### **bookingDateFrom**

The booking date is greater than or equal this date

#### **bookingDateTo**

The booking date is less than or equal this date

#### **itineraryIds**

Array of itinerary ids (AKA Viator Reference) to check for e.g. [1234657,2345267,3245154]

#### **itemIds**

Array of item ids (AKA Viator Item Reference) to check for e.g. [1234657,2345267,3245154]

#### **distributorRefs**

Array of partner-defined distributor references e.g. ["ref1","ref2","ref3"]

#### **distributorItemRefs**

Array of partner-defined distributor item references e.g. ["refitem1","refitem2","refitem3"]

#### **leadFirstName**

The lead traveller's first name

#### **leadSurname**

The lead traveller's surname

#### **test**

Setting test to true will bypass the poll limit on the prelive environment only. The default value for test is false.

#### **Example Result Object**

```

{
  "data": [
    {
      "itineraryId": 3332064,
      "bookingStatus": {
        "type": "CONFIRMED",
        "level": "ITINERARY",
        "failed": false,
        "text": "Confirmed",
        "cancelled": false,
        "status": 3,
        "confirmed": true,
        "amended": false,
        "pending": false
      },
      "bookingDate": "2013-03-25",
      "distributorRef": "Ref20132603_1",
      "itemSummaries": [
        {

```



```

        "itineraryId": 3332064,
        "itemId": 600088886,
        "bookingStatus": {
            "type": "CONFIRMED",
            "level": "ITEM",
            "failed": false,
            "text": "Paid & Confirmed",
            "cancelled": false,
            "status": 1,
            "confirmed": true,
            "amended": false,
            "pending": false
        },
        "travelDate": "2013-12-03",
        "distributorItemRef": "ItemRefA",
        "sortOrder": 0
    },
    "sortOrder": 1
},
{
    "itineraryId": 3332076,
    "bookingStatus": {
        "type": "CONFIRMED",
        "level": "ITINERARY",
        "failed": false,
        "text": "Confirmed",
        "cancelled": false,
        "status": 3,
        "confirmed": true,
        "amended": false,
        "pending": false
    },
    "bookingDate": "2013-03-26",
    "distributorRef": "Ref20132603_5",
    "itemSummaries": [
        {
            "itineraryId": 3332076,
            "itemId": 600088907,
            "bookingStatus": {
                "type": "CONFIRMED",
                "level": "ITEM",
                "failed": false,
                "text": "Paid & Confirmed",
                "cancelled": false,
                "status": 1,
                "confirmed": true,
                "amended": false,
                "pending": false
            },
            "travelDate": "2013-12-03",
            "distributorItemRef": "ItemRefA",
            "sortOrder": 0
        }
    ],
    "sortOrder": 2
}
],
"vmid": "221002",
"errorMessage": null,
"errorType": null,
"dateStamp": "2013-03-26T10:25:57+0000",
"errorReference": null,
"errorMessageText": null,
"success": true,
"totalCount": 2,

```



```

    "errorName": null
}

```

### Exceeding the poll limit

You will receive the following error if you exceed the number of calls allowed to the service within the timeframe:

```

{
  "data": null,
  "vmid": "221002",
  "errorMessage": [
    "Access allowed every 30 minutes"
  ],
  "errorType": "EXCEPTION",
  "dateStamp": "2013-03-26T10:28:51+0000",
  "errorReference": "~55315512721712161381352771",
  "errorMessageText": [
    "Access allowed every 30 minutes"
  ],
  "success": false,
  "totalCount": 1,
  "errorName": "PollingDeniedException"
}

```

## 5.5.10 /booking/pricingmatrix

Method: POST

This API retrieves the pricing matrix for tourgrades, product age bands and pax mixes.

### Example JSON Data

```

{
  "productCode": "5261HTLAP",
  "tourGradeCode": "Zone 1",
  "bookingDate": "2013-12-01",
  "currencyCode": "USD",
  "specialReservation": false
}

```

### bookingDate

The date to check for pricing data. This is an optional parameter for a normal product. If the date is not provided then the nearest available date is determined (i.e. not blocked out or unavailable for any reason)

On the Viator mobile website, the pricing matrix information is displayed in the schedule tab.

Pricing matrix example:



### Example Result Object

```

{
  "data": [
    {
      "sortOrder": 1,
      "bookingDate": "2013-12-01",

```



```

    "pricingUnit": "per person",
    "ageBandPrices": [
      {
        "sortOrder": 1,
        "prices": [
          {
            "currencyCode": "USD",
            "sortOrder": 1,
            "price": 85.13,
            "priceFormatted": "$85.13",
            "merchantNetPrice": 0,
            "merchantNetPriceFormatted": "",
            "minNoOfTravellersRequiredForPrice": 1
          }
        ],
        "bandId": 1,
        "minimumCountRequired": 1,
        "maximumCountRequired": 1
      },
      {
        "sortOrder": 2,
        "prices": [
          {
            "currencyCode": "USD",
            "sortOrder": 1,
            "price": 0,
            "priceFormatted": "$0.00",
            "merchantNetPrice": 0,
            "merchantNetPriceFormatted": "",
            "minNoOfTravellersRequiredForPrice": 1
          }
        ],
        "bandId": 2,
        "minimumCountRequired": 0,
        "maximumCountRequired": null
      }
    ]
  },
  {
    "sortOrder": 2,
    "bookingDate": "2013-12-01",
    "pricingUnit": "per person",
    "ageBandPrices": [
      {
        "sortOrder": 1,
        "prices": [
          {
            "currencyCode": "USD",
            "sortOrder": 1,
            "price": 42.56,
            "priceFormatted": "$42.56",
            "merchantNetPrice": 0,
            "merchantNetPriceFormatted": "",
            "minNoOfTravellersRequiredForPrice": 1
          }
        ],
        "bandId": 1,
        "minimumCountRequired": 2,
        "maximumCountRequired": 2
      },
      {
        "sortOrder": 2,
        "prices": [
          {
            "currencyCode": "USD",
            "sortOrder": 1,

```



```

        "price": 0,
        "priceFormatted": "$0.00",
        "merchantNetPrice": 0,
        "merchantNetPriceFormatted": "",
        "minNoOfTravellersRequiredForPrice": 1
    }
],
    "bandId": 2,
    "minimumCountRequired": 0,
    "maximumCountRequired": null
}
]
},
{
    "sortOrder": 3,
    "bookingDate": "2013-12-01",
    "pricingUnit": "per person",
    "ageBandPrices": [
        {
            "sortOrder": 1,
            "prices": [
                {
                    "currencyCode": "USD",
                    "sortOrder": 1,
                    "price": 28.38,
                    "priceFormatted": "$28.38",
                    "merchantNetPrice": 0,
                    "merchantNetPriceFormatted": "",
                    "minNoOfTravellersRequiredForPrice": 1
                }
            ],
            "bandId": 1,
            "minimumCountRequired": 3,
            "maximumCountRequired": 3
        },
        {
            "sortOrder": 2,
            "prices": [
                {
                    "currencyCode": "USD",
                    "sortOrder": 1,
                    "price": 0,
                    "priceFormatted": "$0.00",
                    "merchantNetPrice": 0,
                    "merchantNetPriceFormatted": "",
                    "minNoOfTravellersRequiredForPrice": 1
                }
            ],
            "bandId": 2,
            "minimumCountRequired": 0,
            "maximumCountRequired": null
        }
    ]
}
],
    "vmid": "221001",
    "errorMessage": null,
    "errorType": null,
    "dateStamp": "2013-06-04T17:08:08+0000",
    "totalCount": 3,
    "errorReference": null,
    "errorMessageText": null,
    "success": true,
    "errorName": null
}

```



## Description of elements in JSON Return object

Object	Element	Type	Comments	To be viewed by consumer	Required field?
<b>data</b>		object	Main response object	N	Y
	sortOrder	smallint	The order in which to show the pricing.	Y	Y
	bookingDate	date	Booking date criteria	Y	Y
	pricingUnit	varchar	The unit the pricing is based on. Currently, only "per person" is supported.	Y	Y
<b>ageBand Prices</b>		object	The available age bands and their pricing based on the criteria.	N	Y
	sortOrder	smallint	The order in which the band is to be shown in on the front end.	Y	Y
	bandId	int	The bandId the data is referring to. The corresponding band description (e.g. Adult, Infant etc) and age range (e.g. from 12 to 99) is available in /product	N	Y
	minimumCountRequired	int	The minimum number of pricing units that apply to these prices		Y
	maximumCountRequired	int	The maximum number of pricing units that apply to these prices		Y
<b>prices</b>		object	The pricing available for the age band, based on the min and max count required	Y	Y
	currencyCode	varchar	The currency of the pricing	Y	Y
	sortOrder	smallint	the order the pricing is to be shown within the bandId	Y	Y
	price	decimal	The price, in decimal format.	N	Y
	priceFormatted	varchar	The price, represented in the currency selected and with two decimal places where applicable.	Y	Y
	merchantNetPrice	decimal	The merchant net, in decimal format. This will always be 0 for retail API partners.	N	Y
	merchantNetPriceFormatted	varchar	the merchant net, in the currency selected and formatted according to the currency. It will always be a blank value "" for retail API partners.	N	Y
	minNoOfTravellersRequiredForPrice	int	The number of units that the pricing applies to. For instance, a minNoOfTravellersRequiredForPrice of 3 means that the price is for 3 people.	Y	Y





### 5.5.11 /booking/voucher

Method: GET

Retrieve the voucher details of either a complete itinerary or an item. The data returned is in HTML and can be wrapped in a header and/or footer.

#### Sample URL Parameters

itineraryId=3299307&leadLastName=DP&itemId=600033670  
or  
voucherKey=3299307:93c7f36a56b18ba1068787ba7fb7988da5c8ad08db77604110141ff21498603e:600033670

#### **voucherKey**

Use either the voucherKey or the three separate parameters.

If voucherKey is provided and other parameters are provided then the voucherKey overrides the other parameters

The voucherKey is obtained from mybookings or when you make a booking.

#### **fullHTML**

Optional parameter. Defaults to false. If true, the full HTML including doc type, html and head tags are returned, otherwise a HTML div fragment is returned.

#### **mobileVoucher**

Optional parameter. Defaults to true. If true, the mobile (cut down) voucher HTML is returned, else the full voucher HTML is returned and fullHTML is ignored.

This field should only be enabled for products that have a voucherOption of VOUCHER\_E or VOUCHER\_ID\_ONLY (e-vouchers available for both options). Do not enable mobileVouchers for paper vouchers (voucherOption of VOUCHER\_PAPER\_ONLY) as no barcode is returned.

The voucher information is available in the responses for /product, /booking/book, /booking/pastbooking, /booking/mybookings. It is also displayed under the redemption Info heading in this service.

#### **Voucher example**





- **Date:** Saturday September 29, 2012
- **Time:** 9:00am
- **Lead Traveler:** David
- **Number of Travelers:** 2 Adults
- **Booking Reference:** VIATOR550002557
- **Product Code:** 2800BMC
- **Confirmation Details :** BMC
- **Location**

Pier 83, West 42nd Street and 12th Avenue, on the Hudson River  
You must exchange your voucher for a boarding pass at the Circle Line  
Ticket Window.

**By Subway:**

Take the 1, 2, 3, 7, 9, A, C, E, N, or R train to Times Square 42nd Street; train B, D, F, or Q to 5th Avenue and 42nd Street, or 4, 5, 6 train to 42nd Street Grand Central Station. Follow directions below for bus.

From 42nd Street take the M42 "Crosstown 42nd Street Pier" bus directly to the Circle Line Pier, or take the M42 "Crosstown Javits Center" bus to the 11th Avenue stop and walk one block west. From 49th Street take the M50 "Crosstown 42nd Street Pier" bus directly to the Circle Line pier.

From George Washington Bridge, Westchester or Connecticut: Take the Henry Hudson Parkway (Route 9A), go south to West 43rd Street. Make a right at 43rd Street to Pier 83.

[illegible]

### 5.5.12 /booking/voucher/email

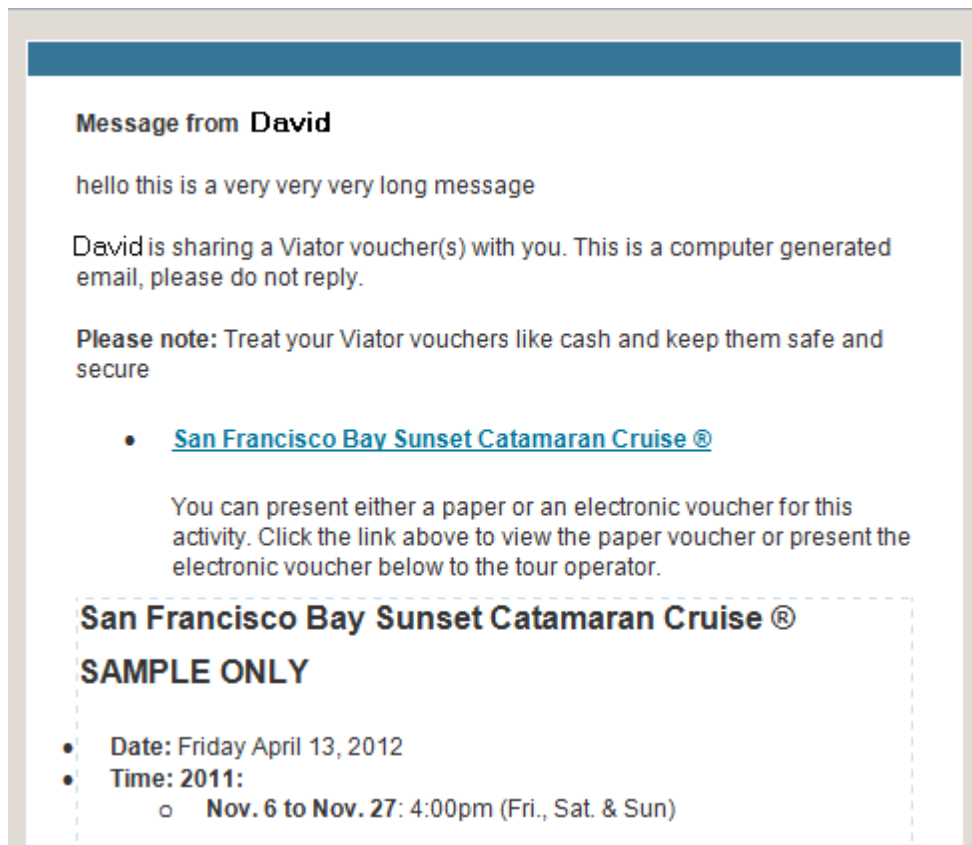
This API will email a voucher to the email specified.

```
{
  itineraryId:3299307,
  itemId: 600033670,
  leadLastName: "DP",
  voucherKey: null,
  name:"David",
  email:"DavidT@viator.com",
  subject:"Tour vouchers from David",
  message:"hello this is a very very very long message"
}
```

Optional. If an itemId is passed (not null and not 0), then only the Individual item voucher gets sent.



## Email example



### Example Result Object

```
{
  data: "Done",
  vmid: "221001",
  errorMessage: null,
  errorType: null,
  dateStamp: "2012-04-13T15:24:12+0000",
  success: true,
  errorReference: null,
  errorMessageText: null,
  totalCount: 1,
  errorName: null
}
```

### 5.5.13 /booking/voucher/fax

Method: POST

This API will fax a voucher to the fax number specified.

### Example JSON Data

```
{
  itineraryId: 3299307,
  itemId: 600033670,
  leadLastName: "DP",
  voucherKey: null,
  countryCode: "61",
  areaCode: "02",
```



```
    faxNumber:"82195499",
    to:"Mary",
    subject:"Hi from David",
    message:"Here are all your tour vouchers for your trip"
}
```

Either pass in `itineraryId`, `leadLastName` and optionally `itemId` or the `voucherKey`.

#### **`voucherKey`**

Optional. If a value is provided, it will override `itineraryId`, `itemId` and `leadLastName`.

#### **`itemId`**

Optional. If an `itemId` is passed (not null and not 0), then only the Individual item voucher gets faxed.

### **Example Result Object**

```
{
  "data": "Done",
  "vmid": "221001",
  "errorMessage": null,
  "errorType": null,
  "dateStamp": "2012-04-13T16:13:40+0000",
  "success": true,
  "errorReference": null,
  "errorMessageText": null,
  "totalCount": 1,
  "errorName": null
}
```

## **5.5.14 /booking/pastbooking**

Method: GET

Get the details of a single specific past booking based on the `voucherKey`, `itineraryId`, or `itemId` passed.

### **Sample URL Parameters**

```
email=john.doe@viator.com&itineraryId=3299307
or
email=john.doe@viator.com&itemId=600033670
or
voucherKey=3299307:db87465199ec5c70f9eb56dd5ad0482e6eb86298f9a7c51283a223c23bbde552:
600033670
```

#### **`email`**

The email address passed must match the email address associated to the itinerary

#### **Departure Details**

Departure details such as the `departurePoint`, `departurePointAddress` and `departurePointDirections` is included in the response. These fields may contain HTML escape characters such as `&`; and special characters that are escaped by a backslash. Ensure that these fields are parsed after receiving the response as it will cause your JSON to be invalid.

### **Example Result Object**



```

{
  "errorReference": null,
  "data":
  {
    "sortOrder": 0,
    "rulesApplied": [ ],
    "bookingStatus":
    {
      "status": 3,
      "text": "Confirmed",
      "type": "CONFIRMED",
      "level": "ITINERARY",
      "confirmed": true,
      "pending": false,
      "amended": false,
      "cancelled": false,
      "failed": false
    },
    "itemSummaries":
    [
      {
        "sortOrder": 0,
        "rulesApplied": [ ],
        "bookingStatus":
        {
          "status": 1,
          "text": "Paid & Confirmed",
          "type": "CONFIRMED",
          "level": "ITEM",
          "failed": false,
          "confirmed": true,
          "amended": false,
          "pending": false,
          "cancelled": false
        },
        "travellerAgeBands":
        [
          {
            "sortOrder": 0,
            "count": 2,
            "pluralDescription": "Adults",
            "description": "Adult",
            "ageBandId": 1
          }
        ],
        "voucherKey": "1000308169:55ec7b23dab801ef1a152c6d63f0df0b5941966d565a0606a7f79",
        "voucherURL": "https://prelive.partner.viator.com/voucher.jspa?PUID=10100&setLo",
        "voucherRequirements": "You must present a paper voucher for this tour. We will",
        "productPulledDown": false,
        "merchantCancellable": false,
        "productWidgetList": null,
        "savingAmount": 0,
        "vouchers": null,
        "passbooks": null,

```



```

        "termsAndConditions": null,
        "itineraryId": 1000308169,
        "productCode": "2065CPT",
        "tourGradeCode": "1DAY",
        "distributorItemRef": null,
        "languageServicesLanguageCode": "en",
        "travelDate": "2015-09-03",
        "price": 35.28,
        "leadTravellerSurname": "Test",
        "barcodeOption": "tour",
        "barcodeType": "code128",
        "destId": 318,
        "voucherOption": "VOUCHER_PAPER_ONLY",
        "productTitle": "City Sightseeing Cape Town Hop-On Hop-Off Tour",
        "itemId": 700179549,
        "obfsId": 84777,
        "departurePoint": "Tour starts at V&A Waterfront, outside the Two Oceans Aqu",
        "departurePointAddress": "",
        "departurePointDirections": "",
        "leadTravellerTitle": "Mr",
        "leadTravellerFirstname": "Homer",
        "lastRetailPrice": 35.28,
        "bookingEngineId": "UF",
        "priceFormatted": "$35.28",
        "savingAmountFormatted": "$0.00",
        "merchantNetPrice": 0,
        "merchantNetPriceFormatted": "",
        "currencyCode": "USD",
        "lastRetailPriceFormatted": "$35.28",
        "departsFrom": "Cape Town, South Africa",
        "tourGradeDescription": "1-Day Bus Tour (1DAY)",
        "hoursConfirmed": 0,
        "priceUSD": 35.28
    }
},
    "voucherURL": "https://prelive.partner.viator.com/voucher.jspa?PUID=10100&setLocale=en&",
    "voucherKey": "1000308169:55ec7b23dab801ef1a152c6d63f0df0b5941966d565a0606a7f7918fb4349",
    "bookerEmail": "jocelyn@viator.com",
    "itineraryId": 1000308169,
    "exchangeRate": 1,
    "distributorRef": null,
    "totalPrice": 35.28,
    "bookingDate": "2015-06-10",
    "totalPriceFormatted": "$35.28",
    "totalPriceUSD": 35.28,
    "hasVoucher": true,
    "userId": null,
    "currencyCode": "USD"
},
    "dateStamp": "2015-06-10T00:20:28+0000",
    "errorType": null,
    "errorMessage": null,
    "errorName": null,
    "success": true,
    "totalCount": 1,
    "vmid": "321004",
    "errorMessageText": null
}
}

```

### 5.5.15 /booking/mybookings

Method: GET



Get a user's bookings with travel dates in the future. This service can also be used to check the status

of a booking.

### Sample URL Parameters

```
sessionId=xxx
or
voucherKey=xxx
or
email=terry.smith@viator.com&lastCCFourDigits=4242
or
email=terry.smith@viator.com&itineraryItemId=3299307
```

Either provide a sessionId for all bookings related to a user account  
 or provide a voucher key  
 or provide an email address and the last four digits of the cc number and get all associated bookings  
 or provide an email address with either an item id or itinerary id  
 In that order

For Failed items show this text:

*The booking has failed either because this tour or activity was not available or there was a technical issue. Please contact Customer Service if you need more information.*

See also [Booking and Item Status List](#)

### Departure Details

Departure details such as the departurePoint, departurePointAddress and departurePointDirections is included in the response. These fields may contain HTML escape characters such as & and special characters that are escaped by a backslash. Ensure that these fields are parsed after receiving the response as it will cause your JSON to be invalid.

### Example Result Object

```
{
  "errorReference": null,
  "data":
  [
    {
      "sortOrder": 1,
      "rulesApplied": [ ],
      "bookingStatus":
      {
        "status": 3,
        "text": "Confirmed",
        "type": "CONFIRMED",
        "level": "ITINERARY",
        "confirmed": true,
        "pending": false,
        "amended": false,
        "cancelled": false,
        "failed": false
      },
      "itemSummaries":
```





```
[
  {
    "sortOrder": 0,
    "rulesApplied": [ ],
    "bookingStatus":

    {
      "status": 1,
      "text": "Paid & Confirmed",
      "type": "CONFIRMED",
      "level": "ITEM",
      "failed": false,
      "confirmed": true,
      "amended": false,
      "pending": false,
      "cancelled": false
    },
    "travellerAgeBands":
    [
      {
        "sortOrder": 0,
        "count": 2,
        "pluralDescription": "Adults",
        "description": "Adult",
        "ageBandId": 1
      }
    ],
    "voucherKey": "1000308169:55ec7b23dab801ef1a152c6d63f0df0b5941966d565a0606a",
    "voucherURL": "https://prelive.partner.viator.com/voucher.jspa?PUID=10100&s",
    "voucherRequirements": "You must present a paper voucher for this tour. We",
    "productPulledDown": false,
    "merchantCancellable": false,
    "productWidgetList": null,
    "savingAmount": 0,
    "vouchers": null,
    "passbooks": null,
    "termsAndConditions": null,
    "itineraryId": 1000308169,
    "productCode": "2065CPT",
    "tourGradeCode": "1DAY",
    "distributorItemRef": null,
    "languageServicesLanguageCode": "en",
    "travelDate": "2015-09-03",
    "price": 35.28,
    "leadTravellerSurname": "Test",
    "barcodeOption": "tour",
    "barcodeType": "code128",
    "destId": 318,
    "voucherOption": "VOUCHER_PAPER_ONLY",
    "productTitle": "City Sightseeing Cape Town Hop-On Hop-Off Tour",
    "itemId": 700179549,
    "obfsId": 84777,
    "departurePoint": "Tour starts at V&A Waterfront, outside the Two Ocean",
    "departurePointAddress": "",
    "departurePointDirections": "",
    "leadTravellerTitle": "Mr",
    "leadTravellerFirstname": "Homer",
    "lastRetailPrice": 35.28,
    "bookingEngineId": "VF",
    "priceFormatted": "R35.28",
    "savingAmountFormatted": "$0.00",
  }
]
```



```

        "merchantNetPrice": 0,
        "merchantNetPriceFormatted": "",
        "currencyCode": "USD",
        "lastRetailPriceFormatted": "$35.28",
        "departsFrom": "Cape Town, South Africa",
        "tourGradeDescription": "1-Day Bus Tour (1DAY)",
        "hoursConfirmed": 0,
        "priceUSD": 35.28
    },
    ],
    "voucherURL": "https://prelive.partner.viator.com/voucher.jsps?PUID=10100&setLocale",
    "voucherKey": "1000308169:55ec7b23dab801ef1a152c6d63f0df0b5941966d565a0606a7f7918fb",
    "bookerEmail": "jocelyn@viator.com",
    "itineraryId": 1000308169,
    "exchangeRate": 1,
    "distributorRef": null,
    "totalPrice": 35.28,
    "bookingDate": "2015-06-10",
    "totalPriceFormatted": "$35.28",
    "totalPriceUSD": 35.28,
    "hasVoucher": true,
    "userId": null,
    "currencyCode": "USD"
}
},
"dateStamp": "2015-06-10T00:22:23+0000",
"errorType": null,
"errorMessage": null,
"errorName": null,
"success": true,
"totalCount": 1,
"vmid": "321004",
"errorMessageText": null
}
}

```

### 5.5.16 /booking/mybookings/message

Method: POST

This API will send a booking message to customer service. A confirmation email will be sent to the email address associated with the booking.

#### Example JSON Data

```

{
  "itineraryId": 3299307,
  "itemId": 600033670,
  "type": "OTHER",
  "message": "Voucher enquiry",
  "otherQuestion": "This question will be overwritten with the question set at otherQuestionId",
  "otherQuestionId": "VOUCHER_PROBLEM",
  "respondInFL": false
}

```

#### sessionId

Partners can either omit the sessionId or leave blank.

#### type

Valid message types are:

AMEND



CANCEL  
OTHER  
REFUND

Partners must specify type of "OTHER"

#### **otherQuestionId**

If specified will override the otherQuestion with a premapped question.

ON\_REQUEST  
GENERAL\_INFO  
VOUCHER\_PROBLEM  
TECH\_PROBLEM  
INCORRECT\_DETAILS  
FAX\_MOBILE  
MOBILE\_UPDATES  
GENERAL\_OTHER

Partners must include this parameter.

#### **respondInFL**

Only applicable for "PARTNER"/"VML" sites. If true then respond in Foreign Language.

#### **Cancellations**

Example JSON to make a cancellation:

```
{
  "itineraryId": 3299307,
  "itemId": 600033670,
  "type": "CANCEL",
  "message": "Holiday cancelled by the customer"
}
```

It will then go to a queue to be canceled manually by the Viator customer service team.

#### **Example Result Object**

```
{
  "data": "Successfully Submitted booking message",
  "vmid": "221001",
  "errorMessage": null,
  "errorType": null,
  "dateStamp": "2012-04-16T13:03:54+0000",
  "success": true,
  "errorReference": null,
  "errorMessageText": null,
  "totalCount": 1,
  "errorName": null
}
```

\$bookingQuestionList["[ON-REQUEST]NotReceived"] = 'I have not yet received confirmation for my tour/activity';

\$bookingQuestionList["[ON-REQUEST]incorrectDetails"] = 'My tour is showing incorrect details';

\$bookingQuestionList["[FAX-MOBILE]newConfirmation"] = 'I need my confirmation and/or Voucher faxed to my hotel';

\$bookingQuestionList["[VOUCHER-PROBLEM]cannotPrint"] = 'I cannot access or print my Viator Voucher(s)';



```
$bookingQuestionList["[FAX-MOBILE]cannotAccessEmail"] = 'I can not access email; call my cell/  
mobile with updates';  
$bookingQuestionList["[GENERAL-INFO]otherQuestion"] = 'Other booking question';
```

## 5.6 Support APIs

The links to the terms and conditions, FAQ and customer care page is available to API partners.

### 5.6.1 /support/terms

Method: GET

No parameters

This API returns the URL for the terms and conditions. The URL is not partner specific so any customization on the look and feel of the partner's Viator white label site will not be shown.

Currently only one field is returned - url. This contains the fully qualified URL to the terms and conditions.

Please note that this information is normally contained in a pop up.

#### Example Result Object

```
{  
  
  "errorReference": null,  
  "data": {  
    "url": "http://prelive.partner.viator.com/popups/terms.jspa"  
  },  
  "dateStamp": "2013-09-03T16:33:10+0000",  
  "errorType": null,  
  "errorMessage": null,  
  "errorName": null,  
  "success": true,  
  "totalCount": 1,  
  "vmid": "221002",  
  "errorMessageText": null  
}
```

### 5.6.2 /support/faq

Method: GET

No parameters

This API returns the URL to the FAQ page that the partner can link to within their app / website. The FAQ page is hosted on the partner's white label site hence any customisation made on the white label site will be displayed.

Currently only one field is returned - url. This contains the fully qualified URL to the partner's FAQ page.

#### Example Result Object

```
{  
  
  
```

```

    "errorReference": null,
    "data": {
      "url": "http://qa.prelive.partner.viator.com/en/10627/faq"
    },
    "dateStamp": "2013-09-03T16:33:35+0000",
    "errorType": null,
    "errorMessage": null,
    "errorName": null,
    "success": true,
    "totalCount": 1,
    "vmid": "221002",
    "errorMessageText": null
  }
}

```

### 5.6.3 /service/support/customer-care

Method: GET

No parameters

This API returns the URL to the customer care page that the partner can link to within their app / website. The customer care page is hosted on the partner's white label site hence any customisation made on the white label site will be displayed.

Currently only one field is returned - url. This contains the fully qualified URL to the partner's customer care page.

#### Example Result Object

```

{
  "errorReference": null,
  "data": {
    "url": "http://prelive.partner.viator.com/en/10627/customer-care"
  },
  "dateStamp": "2013-09-03T16:33:53+0000",
  "errorType": null,
  "errorMessage": null,
  "errorName": null,
  "success": true,
  "totalCount": 1,
  "vmid": "221002",
  "errorMessageText": null
}

```

## 6 Sample Code

### 6.1 PHP Sample Code

#### 6.1.1 POST Request

This is a sample PHP POST API request to get the detailed availability of a product for a month.

```
function getAvailability ( $productCode, $month, $year, $currencyCode, $ageBandIdQtyArray )
```



```

{
    $url = getServiceURL() . "/service/booking/availability" . "?apiKey=" . getAPIKey();

    // set up an array for the agebands -> qty list
    $agebandQtyList = array();
    foreach ( $ageBandIdQtyArray as $band => $qty )
    {
        $agebandQty = new agebandQty();
        $agebandQty->bandId = $band;
        $agebandQty->count = $qty;

        $agebandQtyList[] = $agebandQty;
    }

    // Sample JSON : {"productCode":"2280AAHT","month":"6", "year":"2011",
                    "currencyCode":"EUR", "ageBands":[{"bandId":1, "count":1}]}

    // add the other parameters
    $data['productCode'] = $productCode;
    $data['month'] = $month;
    $data['year'] = $year;
    $data['currencyCode'] = $currencyCode;
    $data['ageBands'] = $agebandQtyList;

    // encode the $data, post to the service, decode the JSON and return the resulting object
    return json_decode ( do_post_request( $url, json_encode( $data ),
                                         "Content-Type: application/json\n" ) );
}

// THIS IS A HELPER FUNCTION TO WRAP AN HTTP POST
// $data = array(); - this is a key value assoc array to emulate a form being posted...
// $response = do_post_request($url, http_build_query($data) );
function do_post_request($url, $data, $optional_headers = null)
{
    $params = array('http' => array(
        'method' => 'POST',
        'content' => $data
    ));

    if ($optional_headers !== null) {
        $params['http']['header'] = $optional_headers;
    }

    $ctx = stream_context_create($params);
    $fp = @fopen($url, 'rb', false, $ctx);
    if (!$fp) {
        throw new Exception("Problem with $url, $php_errormsg");
    }
    $response = @stream_get_contents($fp);

    if ($response === false) {
        throw new Exception("Problem reading data from $url, $php_errormsg");
    }
    return $response;
}

```

## 6.1.2 GET Request

Here's some sample code for getting the product details.

```

function getProduct ( $code, $currencyCode )
{
    global $prefix, $memcache, $cacheTime;

    // create key for memcache
    $key = $prefix."_".getAPIKey()."_".getProduct."_".$code."_".$currencyCode;

```



```

// get the result from memcache if it's available
$ret = $memcache->get( $key );

// if not available, call the API
if ( $ret === false )
{
    // form the URL
    $url = getServiceURL() . "/service/product" . "?apiKey=" . getAPIKey() .
        "&code=" . $code . "&currencyCode=".$currencyCode;

    // do the GET and decode the results
    $ret = json_decode ( file_get_contents( $url ) );

    // check the return object for a successful request
    if ( $ret->success === true )
    {
        // set the object into memcache for next time
        $memcache->set($key, $ret, false, $cacheTime);
    }
}

// return the data object
return $ret;
}

```

## 6.2 Java Sample Code

### 6.2.1 Java POST Request

This is a sample Java POST API request to get the products in Las Vegas available on the 25th February, 2014

Note that this for Java code with Apache http-client 4.3.2:

```

HttpPost post = new HttpPost("http://prelive.viatorapi.viator.com/service/search/products?apiKey=
post.setHeader("Content-Type", "application/json");

String json = "

{\"startDate\":\"2014-02-25\",\"endDate\":\"2014-02-25\", \"topX\":\"1-5\",\"destId\":\"684\",
";

post.setEntity(new StringEntity(json));
CloseableHttpClient httpclient = HttpClients.createDefault();
CloseableHttpResponse response = httpclient.execute(post);
InputStream content = response.getEntity().getContent();

```

## 7 Appendix



## 7.1 Standard JSON fields

Every service returns a standard set of JSON fields at the end of the JSON response, which indicates if it was processed successfully by the API.

In addition to the success flag, you will also need to check the `errorMessage` values for the status of the response. Success: true and `errorType` / `errorMessage` of null indicates no errors.

### Example JSON - successful:

```
{
  "data": [
  ],
  "vmid": "321001",
  "errorMessage": null,
  "errorType": null,
  "dateStamp": "2013-03-06T19:45:10+0000",
  "errorReference": null,
  "errorMessageText": null,
  "success": true,
  "totalCount": 114,
  "errorName": null
}
```

### Example JSON - unsuccessful:

```
{
  "errorReference": "~5793740141815885188840666",
  "data": null,
  "dateStamp": "2013-09-09T11:29:48+0000",
  "errorType": "EXCEPTION",
  "errorMessage": [
    "* Additional questions missing\n"
  ],
  "errorName": "ValidationException",
  "success": false,
  "totalCount": 1,
  "vmid": "221001",
  "errorMessageText": [
    "* Additional questions missing"
  ]
}
```





Element	Type	Comments	To be viewed by consumer	Required field?
vmid	varchar	The server id that processed the service	N	Y
errorMessage	varchar	The error message in HTML	N	Y
errorType	varchar	Type of error: EXCEPTION	N	Y
dateStamp	datetime	timestamp of the response	N	Y
errorReference	varchar	The error reference is logged for future reference	N	Y
errorMessageText	varchar	The textual version of the error message	Y, if an error has occurred	Y
success	boolean	True if the JSON request was successful with no errors, false if an error was encountered	N	Y
totalCount	smallint	The number of results returned. 1 is the minimum.	Y, if displaying the number of results found in a search etc.	Y
errorName	varchar	The name of the error type	N	Y

## 7.2 Country Codes

AF|Afghanistan  
 AL|Albania  
 DZ|Algeria  
 AS|American Samoa  
 AD|Andorra  
 AO|Angola  
 AI|Anguilla  
 AQ|Antarctica  
 AG|Antigua and Barbuda  
 AR|Argentina  
 AM|Armenia  
 AW|Aruba  
 AU|Australia  
 AT|Austria  
 AZ|Azerbaijan  
 BS|Bahamas  
 BH|Bahrain  
 BD|Bangladesh  
 BB|Barbados  
 BY|Belarus  
 BE|Belgium  
 BZ|Belize  
 BJ|Benin  
 BM|Bermuda  
 BT|Bhutan  
 BO|Bolivia  
 BA|Bosnia Herzegovina  
 BW|Botswana  
 BR|Brazil  
 BN|Brunei



BG|Bulgaria  
BF|Burkina Faso  
BI|Burundi  
KH|Cambodia  
CM|Cameroon  
CA|Canada  
CV|Cape Verde  
KY|Cayman Islands  
CF|Central Africa  
TD|Chad  
CL|Chile  
CN|China  
CX|Christmas Island  
CC|Cocos (Keeling) Islands  
CO|Colombia  
KM|Comoros  
CK|Cook Islands  
CR|Costa Rica  
CI|Cote D'Ivoire  
HR|Croatia  
CY|Cyprus  
CZ|Czech Republic  
DK|Denmark  
DJ|Djibouti  
DM|Dominica  
DO|Dominican Republic  
EC|Ecuador  
EG|Egypt  
SV|El Salvador  
GQ|Equatorial Guinea  
ER|Eritrea  
EE|Estonia  
ET|Ethiopia  
FK|Falkland Island  
FO|Faroe Islands  
FJ|Fiji  
FI|Finland  
FR|France  
GF|French Guiana  
PF|French Polynesia  
GA|Gabon  
GM|Gambia  
GE|Georgia  
DE|Germany  
GH|Ghana  
GI|Gibraltar  
GR|Greece  
GL|Greenland  
GD|Grenada  
GP|Guadeloupe  
GU|Guam  
GT|Guatemala  
GN|Guinea  
GW|Guinea Bissau  
GY|Guyana  
HT|Haiti  
HN|Honduras



HK|Hong Kong  
HU|Hungary  
IS|Iceland  
IN|India  
ID|Indonesia  
IQ|Iraq  
IE|Ireland  
IL|Israel  
IT|Italy  
JM|Jamaica  
JP|Japan  
JO|Jordan  
KZ|Kazakhstan  
KE|Kenya  
KI|Kiribati  
KW|Kuwait  
KG|Kyrgyzstan  
LA|Lao People's Democratic Republic  
LV|Latvia  
LB|Lebanon  
LS|Lesotho  
LR|Liberia  
LY|Libyan Arab Jamahiriya  
LI|Liechtenstein  
LT|Lithuania  
LU|Luxembourg  
MO|Macau  
MK|Macedonia  
MG|Madagascar  
MW|Malawi  
MY|Malaysia  
MV|Maldives  
ML|Mali  
MT|Malta  
MQ|Martinique  
MR|Mauritania  
MU|Mauritius  
YT|Mayotte  
MX|Mexico  
FM|Micronesia  
MD|Moldova  
MC|Monaco  
MN|Mongolia  
MS|Montserrat  
MA|Morocco  
MZ|Mozambique  
NA|Namibia  
NR|Nauru  
NP|Nepal  
NL|Netherlands  
AN|Netherlands Antilles  
KN|Nevis- St Kitts  
NC|New Caledonia  
NZ|New Zealand  
NI|Nicaragua  
NE|Niger  
NG|Nigeria



NU|Niue  
NF|Norfolk Island  
KP|North Korea  
MP|Northern Mariana Islands  
NO|Norway  
OM|Oman  
PK|Pakistan  
PW|Palau  
PS|Palestinian Territory, Occupied  
PA|Panama  
PG|Papua New Guinea  
PY|Paraguay  
PE|Peru  
PH|Philippines  
PN|Pitcairn  
PL|Poland  
PT|Portugal  
PR|Puerto Rico  
QA|Qatar  
RE|Reunion  
RO|Romania  
RU|Russian Federation  
RW|Rwanda  
SH|Saint Helena  
LC|Saint Lucia  
SM|San Marino  
ST|Sao Tome and Principe  
SA|Saudi Arabia  
SN|Senegal  
YU|Serbia and Montenegro  
SC|Seychelles  
SL|Sierra Leone  
SG|Singapore  
SK|Slovakia  
SI|Slovenia  
SB|Solomon Islands  
SO|Somalia  
ZA|South Africa  
KR|South Korea  
ES|Spain  
LK|Sri Lanka  
PM|St Pierre Miquelon  
VC|St Vincent and Grenadines  
SR|Suriname  
SZ|Swaziland  
SE|Sweden  
CH|Switzerland  
SY|Syria  
TW|Taiwan  
TJ|Tajikistan  
TZ|Tanzania  
TH|Thailand  
TL|Timor-Leste  
TG|Togo  
TK|Tokelau  
TO|Tonga  
TT|Trinidad and Tobago



TN|Tunisia  
TR|Turkey  
TM|Turkmenistan  
TC|Turks and Caicos Islands  
TV|Tuvalu  
UG|Uganda  
UA|Ukraine  
AE|United Arab Emirates  
GB|United Kingdom  
UY|Uruguay  
UM|US Minor Outlying Islands  
US|United States of America  
UZ|Uzbekistan  
VU|Vanuatu  
VE|Venezuela  
VN|Vietnam  
VG|Virgin Islands-British  
VI|Virgin Islands-US  
WF|Wallis and Futuna Islands  
WS|Western Samoa  
YE|Yemen Republic  
ZM|Zambia  
ZW|Zimbabwe

## 7.3 USA States

AL|Alabama  
AK|Alaska  
AZ|Arizona  
AR|Arkansas  
CA|California  
CO|Colorado  
CT|Connecticut  
DE|Delaware  
DC|District of Columbia  
FL|Florida  
GA|Georgia  
HI|Hawaii  
ID|Idaho  
IL|Illinois  
IN|Indiana  
IA|Iowa  
KS|Kansas  
KY|Kentucky  
LA|Louisiana  
ME|Maine  
MD|Maryland  
MA|Massachusetts  
MI|Michigan  
MN|Minnesota  
MS|Mississippi  
MO|Missouri  
MT|Montana  
NE|Nebraska  
NV|Nevada  
NH|New Hampshire



NJ|New Jersey  
NM|New Mexico  
NY|New York  
NC|North Carolina  
ND|North Dakota  
OH|Ohio  
OK|Oklahoma  
OR|Oregon  
PA|Pennsylvania  
RI|Rhode Island  
SC|South Carolina  
SD|South Dakota  
TN|Tennessee  
TX|Texas  
UT|Utah  
VT|Vermont  
VA|Virginia  
WA|Washington  
WV|West Virginia  
WI|Wisconsin  
WY|Wyoming

## 7.4 Canada Provinces

Alberta|Alberta  
British Columbia|British Columbia  
Manitoba|Manitoba  
New Brunswick|New Brunswick  
Newfoundland and Labrador|Newfoundland  
Northwest Territories|Northwest Territories  
Nova Scotia|Nova Scotia  
Nunavut|Nunavut  
Ontario|Ontario  
Prince Edward Island|Prince Edward Island  
Quebec|Quebec  
Saskatchewan|Saskatchewan  
Yukon|Yukon Territory

## 7.5 Australian States

ACT|Australian Capital Territory  
NSW|New South Wales  
NT|Northern Territory  
QLD|Queensland  
SA|South Australia  
TAS|Tasmania  
VIC|Victoria  
WA|Western Australia

## 7.6 Booking and Item Status List

```
$bookingStatus['STATE_Waiting'] = 0;  
$bookingStatus['STATE_BeingBooked'] = 2;  
booked  
$bookingStatus['STATE_Confirmed'] = 3;
```



```
// Itinerary is waiting to be booked  
// Itinerary is in process of being  
// Itinerary has been booked. No
```

more change processing is allowed	
\$bookingStatus['STATE_PendingBooking'] = 4;	// Itinerary is in the process of
handling a delayed / offline (asynchronous) booking	
\$bookingStatus['STATE_Failed'] = 5;	// Itinerary has failed the booking
process	
\$bookingStatus['STATE_BookingSet'] = 6;	// Itinerary has paused in the
ActionBookSet process	
\$bookingStatus['STATE_BeingBookedSet'] = 2;	// Itinerary is in process of being
booked with the BookSet command.	
\$bookingStatus['STATE_BeingBookedCommit'] = 8;	// Itinerary is in process of being
booked with the BookCommit command	
\$bookingStatus['STATE_BeingCancelled'] = 9;	// Itinerary is in process of being
cancelled.	
\$bookingStatus['STATE_Cancelled'] = 10;	// Itinerary has been cancelled.
\$bookingStatus['STATE_PendingCancellation'] = 11;	// Itinerary is in the process of
handling a delayed / offline (asynchronous) cancellation.	
\$bookingStatus['STATE_Amended'] = 12;	// Itinerary has been amended
\$bookingStatus['STATE_PendingAmendment'] = 13;	// Itinerary is in process of
amending a booking	
\$itemStatus['BOOKINGREFERENCE_Waiting'] = 0;	// This item has not been booked.
Part of an unfinished itinerary.	
\$itemStatus['BOOKINGREFERENCE_Confirmed'] = 1;	// This item has been successfully
booked.	
\$itemStatus['BOOKINGREFERENCE_Unavailable'] = 2;	// This item has been had an
availability check, that came back false.	
\$itemStatus['BOOKINGREFERENCE_Pending'] = 3;	// This item has begun booking, but
has paused in a deferred booking engine.	
\$itemStatus['BOOKINGREFERENCE_Failed'] = 4;	// This item has failed the booking
process.	
\$itemStatus['BOOKINGREFERENCE_Cancelled'] = 5;	// This item has successfully been
cancelled.	
\$itemStatus['BOOKINGREFERENCE_Expired'] = 6;	// This item is now expired
\$itemStatus['BOOKINGREFERENCE_Amended'] = 7;	// This item has been amended
after booking	
\$itemStatus['BOOKINGREFERENCE_Pending_Amend'] = 8;	// This item has a pending
amendment which can be cancelled	

## 7.7 Required Info Questions

Some products have mandatory questions that need to be answered in order to be booked. The questions are returned in service /product.

List of booking questions:

For safety reasons you must enter the weight of **all** passengers individually. Please indicate pounds or kilos.  
 Enter name of your airline  
 Enter your flight number  
 Enter your flight's arrival time. Please indicate AM/PM or use the 24-hour clock.  
 Enter your drop-off hotel  
 Enter your flight's departure time. Please indicate AM/PM or use the 24-hour clock.  
 Enter hotel name for pickup  
 Enter departure date, flight number & departure time  
 Enter arrival station, train number and rail provider  
 Enter your arrival time. Please indicate AM/PM or use the 24-hour clock.

Enter departure station, train number and rail provider  
Enter your departure time. Please indicate AM/PM or use the 24-hour clock.  
Enter your pick-up hotel  
Enter departure date, departure station, train number and rail provider  
Enter address for drop off  
Enter address for pickup  
Enter your ship name  
Enter ship's arrival time and preferred port pick-up time. Indicate AM/PM or use 24-hour clock.  
Enter your arrival airport & flight details, hotel name & address, or railway station  
Enter ship's departure time and preferred port drop off time. Indicate AM/PM or use 24-hour clock.  
Enter your departure airport & flight details, hotel name & address, or railway station  
You must enter the date of birth for all passengers.  
Enter date of birth for all passengers  
Enter passport number for all passengers  
Enter country of issue of passport for all passengers  
Enter passport expiry date for all passengers  
Enter cruise ship name  
Enter your cruise's disembarkation time. Please indicate AM/PM or use the 24-hour clock.  
Enter your boarding time. Please indicate AM/PM or use the 24-hour clock.  
Enter the next stop of your cruise

## 7.8 Supported currency codes

Supported currency codes for partners with multi currency enabled:

USD|US Dollar  
GBP|British Pound  
EUR|Euro  
AUD|Australian Dollar  
CAD|Canadian Dollar  
NZD|New Zealand Dollar

**Note:** Partners will be billed in the currency of the booking

