

# ABH-QA Task implementation

## Dolores Jureta

### 1. Writing test cases

<b>Test Case: 001</b>	<b>Title:</b> Getting a list of places
<b>Precondition:</b>	
User must be logged to skyscanner with Google, Facebook or Github account	
<b>Description:</b>	
Getting list of places by sending valid data	
<b>Steps:</b>	
<ol style="list-style-type: none"><li>1. Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a></li><li>2. Click on Test Endpoints button</li><li>3. Click on Places</li><li>4. Select get List Places</li><li>5. In query enter Stockholm</li><li>6. In country enter UK</li><li>7. In currency enter GBP</li><li>8. In locale enter en-GB</li><li>9. Click on Test Endpoint button</li></ol>	
<b>Expected result:</b>	
Getting list of all places that matches sent query	
<b>Actual result:</b>	
As expected	
<b>Pass/Fail:</b>	
Pass	
<b>Screenshot:</b>	

GET To Skyscanner

Places

GET List Places

Browse Flight Prices

Localisation

GET List markets

GET Currencies

X-RapidAPI-Key

ENUM

98d989ab10msh6046e420b738258p1... ▾

REQUIRED

X-RapidAPI-Host

ENUM

skyscanner-skyscanner-flight-search-v1... ▾

REQUIRED

Required Parameters

query

STRING

Stockholm

REQUIRED

The query string, must be at least 2 characters long.

country

STRING

UK

REQUIRED

The code of the country must be

200 OK

Body

Headers

```

{
  "Places": [
    {
      "0": [...], 6 items
    },
    {
      "1": [...], 6 items
    },
    {
      "2": [...], 6 items
    },
    {
      "3": [...], 6 items
    },
    {
      "4": [...], 6 items
    }
  ]
}
```

<b>Test Case: 002</b>	<b>Title: Getting a list of places</b>
<b>Precondition:</b>	
User must be logged to skyscanner with Google, Facebook or Github account	
<b>Description:</b>	
Getting list places by sending invalid data	
<b>Steps:</b>	
<ol style="list-style-type: none"> <li>Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a></li> <li>Click on Test Endpoints button</li> <li>Click on Places</li> <li>Select get List Places</li> <li>In query enter B</li> <li>In country enter UK</li> <li>In currency enter GBP</li> <li>In locale enter en-GB</li> <li>Click on Test Endpoint button</li> </ol>	
<b>Expected result:</b>	
Expecting to fail because string query must be at least 2 characters, but we sent only one	
<b>Actual result:</b>	
Pass	
<b>Pass/Fail:</b>	

Fail

### Screenshot:

The screenshot displays a REST client interface. On the left, the API endpoint is `GET Browse Flight Prices` with a required parameter `query` set to `B`. The right pane shows a `200 OK` status and a JSON response body. The JSON body contains an array of places, with the first item being Birmingham, UK.

```
{
  "Places": [
    {
      "PlaceId": "BHX-sky",
      "PlaceName": "Birmingham",
      "CountryId": "UK-sky",
      "RegionId": "",
      "CityId": "BIRM-sky",
      "CountryName": "United Kingdom"
    }
  ]
}
```

**Test Case: 003**

**Title:** Getting browse quotes

### Precondition:

User must be logged to skyscanner with Google, Facebook or Github account

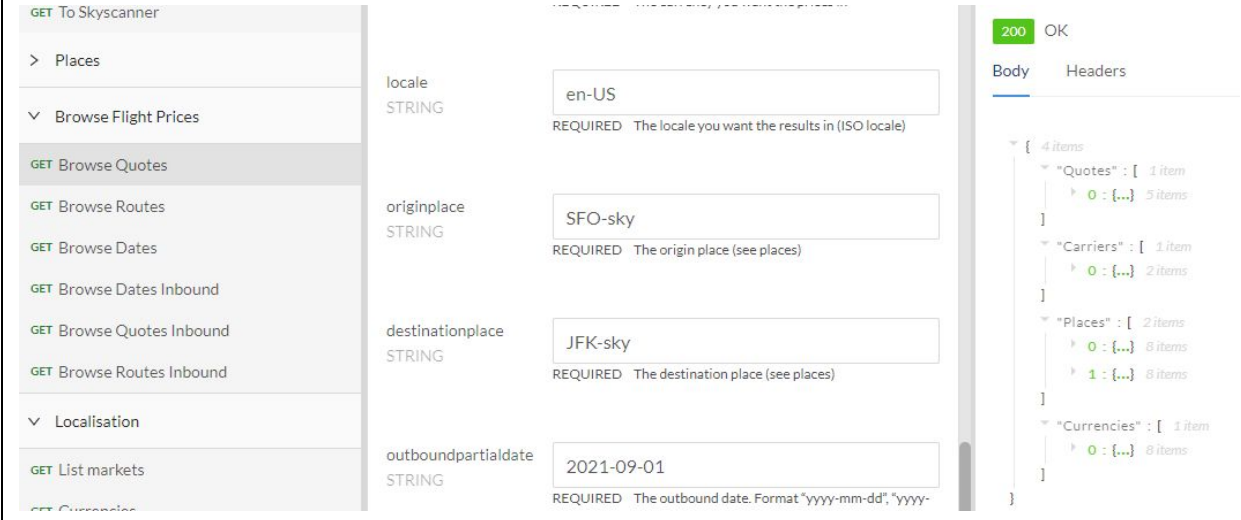
### Description:

Getting list of quotes by sending valid data

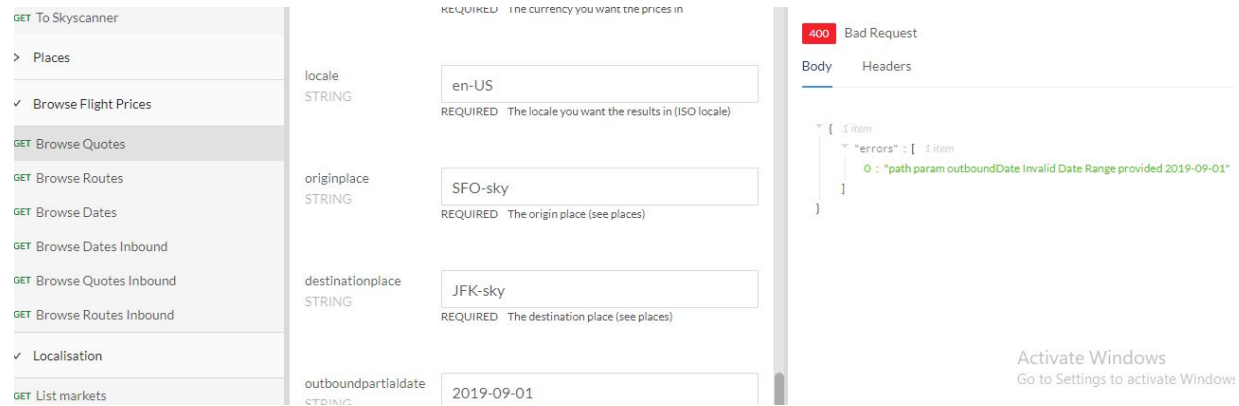
### Steps:

1. Go to site <https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details>
2. Click on Test Endpoints button
3. Click on Browse Flight Prices
4. Select get Browse Quotes
5. In country enter US
6. In currency enter USD
7. In locale enter en-US
8. In originplace enter SFO-sky
9. In destinationplace enter JFK-sky
10. In outboundpartialdate enter 2021-09-01
11. Click on Test Endpoint button

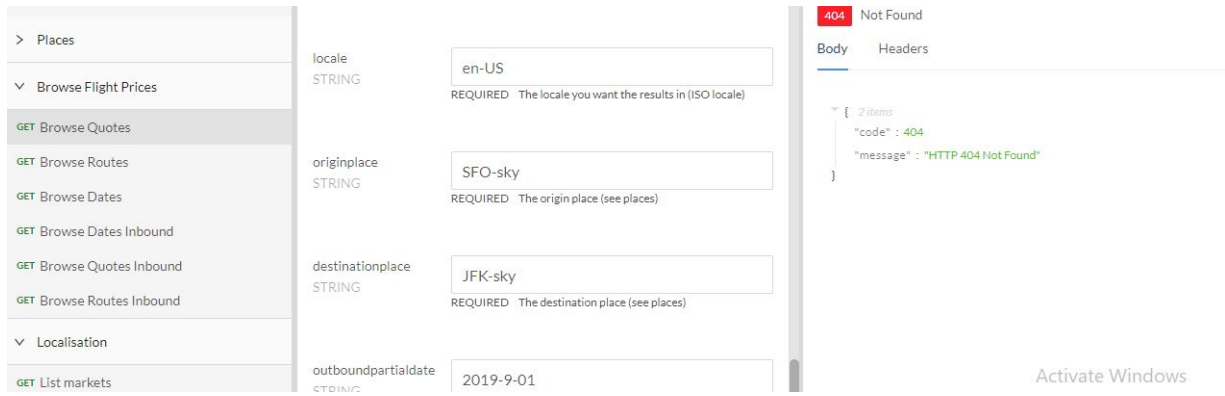
### Expected result:

Getting list of quotes that are cheapest	
<b>Actual result:</b>	
As expected	
<b>Pass/Fail:</b>	
Pass	
<b>Screenshot:</b>	
 <p>The screenshot displays the RapidAPI interface for the Skyscanner API. On the left, a sidebar lists various endpoints, with 'Browse Quotes' selected. The main area shows the configuration for the 'GET Browse Quotes' endpoint. Parameters are set as follows: 'locale' is 'en-US', 'originplace' is 'SFO-sky', 'destinationplace' is 'JFK-sky', and 'outboundpartialdate' is '2021-09-01'. Each parameter has a 'REQUIRED' note. On the right, the response is shown as a 200 OK status with a JSON body. The JSON structure includes 'Quotes' (1 item), 'Carriers' (1 item), 'Places' (2 items), and 'Currencies' (1 item). Each of these top-level objects contains an array of items, with counts shown in brackets (e.g., '0 : [...] 5 items').</p>	

<b>Test Case: 004</b>	<b>Title: Getting browse quotes</b>
<b>Precondition:</b>	
User must be logged to skyscanner with Google, Facebook or Github account	
<b>Description:</b>	
Getting list of browse quotes by sending invalid data	
<b>Steps:</b>	
<ol style="list-style-type: none"> <li>1. Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a></li> <li>2. Click on Test Endpoints button</li> <li>3. Click on Browse Flight Prices</li> <li>4. Select get Browse Quotes</li> <li>5. In country enter US</li> <li>6. In currency enter USD</li> <li>7. In locale enter en-US</li> <li>8. In originplace enter SFO-sky</li> <li>9. In destinationplace enter JFK-sky</li> </ol>	

10. In outboundpartialdate enter 2019-09-01 11. Click on Test Endpoint button	
<b>Expected result:</b>	
Getting error because date is incorrect	
<b>Actual result:</b>	
As expected	
<b>Pass/Fail:</b>	
Pass	
<b>Screenshot:</b>	
	

<b>Test Case: 005</b>	<b>Title:</b> Getting a list of browse quotes
<b>Precondition:</b>	
User must be logged to skyscanner with Google, Facebook or Github account	
<b>Description:</b>	
Getting list of browse quotes by sending invalid data	
<b>Steps:</b>	
<ol style="list-style-type: none"> <li>1. Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a></li> <li>2. Click on Test Endpoints button</li> <li>3. Click on Browse Flight Prices</li> <li>4. Select get Browse Quotes</li> <li>5. In country enter US</li> <li>6. In currency enter USD</li> <li>7. In locale enter en-US</li> </ol>	

8. In originplace enter SFO-sky 9. In destinationplace enter JFK-sky 10. In outboundpartialdate enter 2019-9-01 11. Click on Test Endpoint button
<b>Expected result:</b>
Getting error because date is incorrect
<b>Actual result:</b>
As expected
<b>Pass/Fail:</b>
Pass
<b>Screenshot:</b>


<b>Test Case: 006</b>	<b>Title:</b> Getting a list of browse routes
<b>Precondition:</b>	
User must be logged to skyscanner with Google, Facebook or Github account	
<b>Description:</b>	
Getting list of browse routes by sending valid data	
<b>Steps:</b>	
<ol style="list-style-type: none"> <li>1. Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a></li> <li>2. Click on Test Endpoints button</li> <li>3. Click on Browse Flight Prices</li> <li>4. Select get Browse Routes</li> <li>5. In country enter US</li> </ol>	

6. In currency enter USD
7. In locale enter en-US
8. In originplace enter SFO-sky
9. In destinationplace enter ORD-sky
10. In outboundpartialdate enter 2021-09-01
11. Click on Test Endpoint button

#### Expected result:

Getting list of data including routes

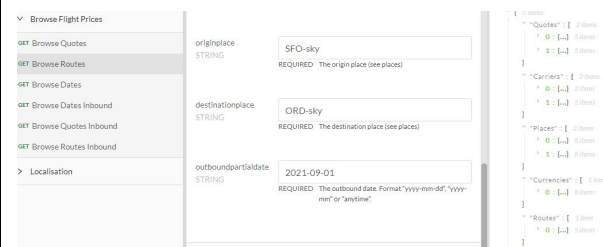
#### Actual result:

As expected

#### Pass/Fail:

Pass

#### Screenshot:



#### Test Case: 007

#### Title: Getting a list of browse routes

#### Precondition:

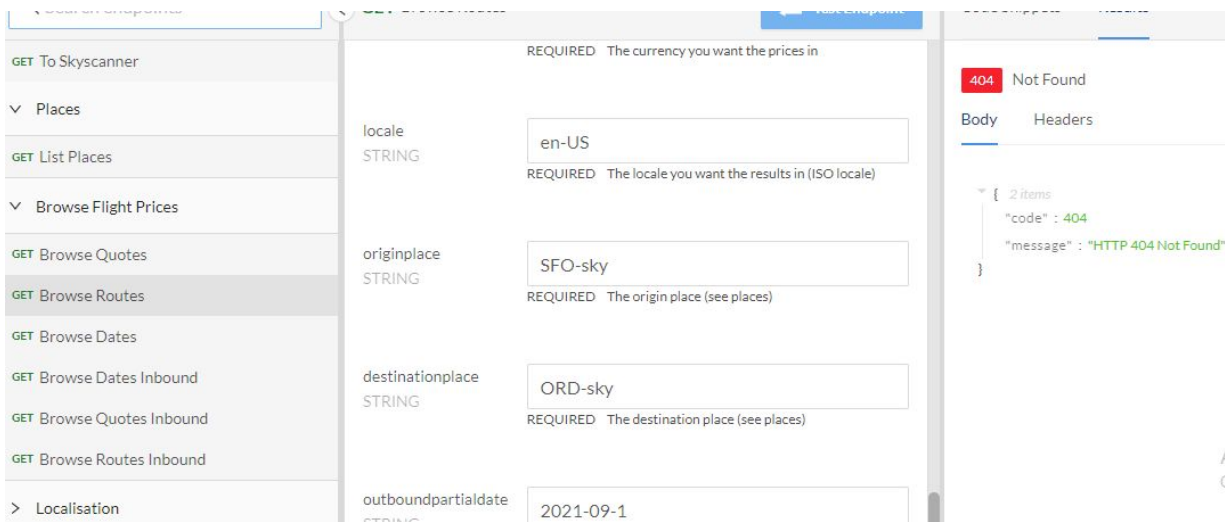
User must be logged to skyscanner with Google, Facebook or Github account

#### Description:

Getting list of browse routes by sending invalid data

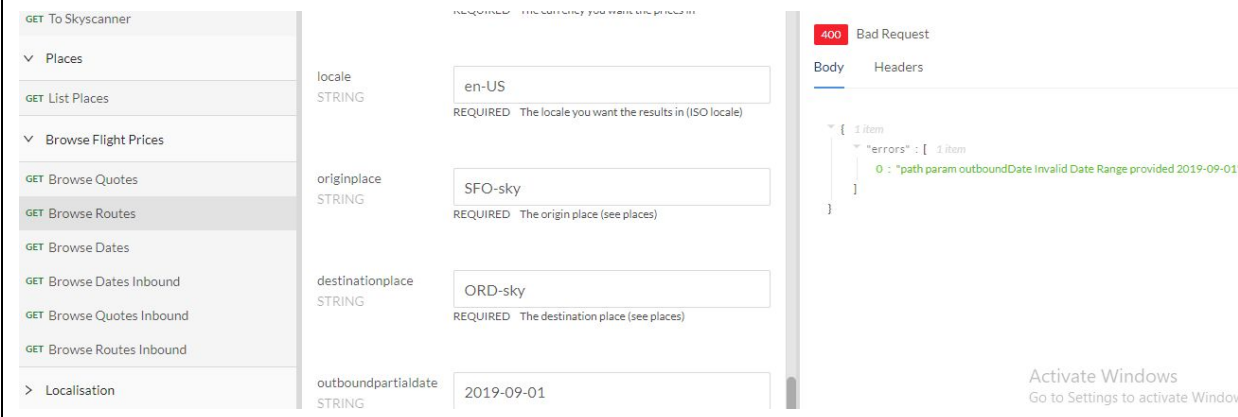
#### Steps:

1. Go to site <https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details>
2. Click on Test Endpoints button
3. Click on Browse Flight Prices
4. Select get Browse Routes
5. In country enter US
6. In currency enter USD
7. In locale enter en-US
8. In originplace enter SFO-sky

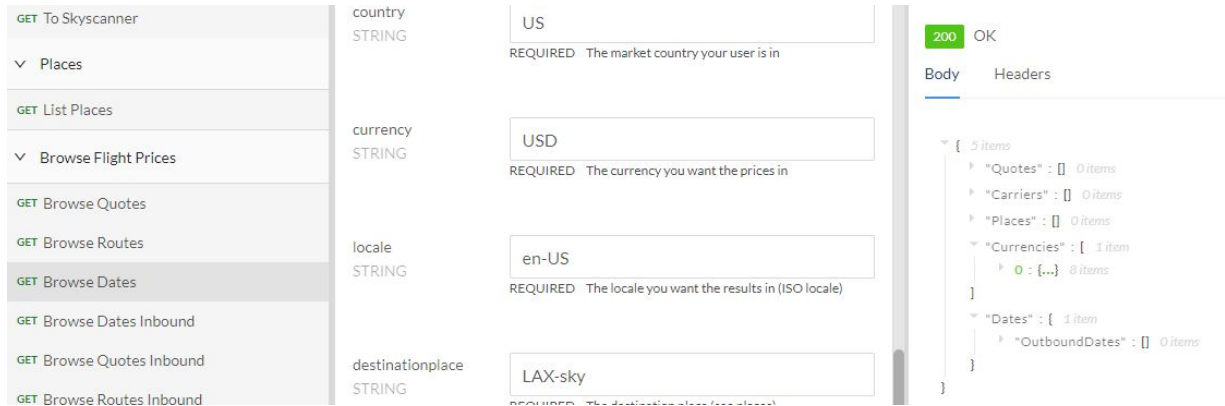
9. In destinationplace enter ORD-sky 10. In outboundpartialdate enter 2019-09-1 11. Click on Test Endpoint button	
<b>Expected result:</b>	
Getting error because date is incorrect	
<b>Actual result:</b>	
As expected	
<b>Pass/Fail:</b>	
Pass	
<b>Screenshot:</b>	
	

<b>Test Case: 008</b>	<b>Title:</b> Getting a list of browse routes
<b>Precondition:</b>	
User must be logged to skyscanner with Google, Facebook or Github account	
<b>Description:</b>	
Getting list of data including routes by sending invalid data	
<b>Steps:</b>	
1. Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a> 2. Click on Test Endpoints button 3. Click on Browse Flight Prices	

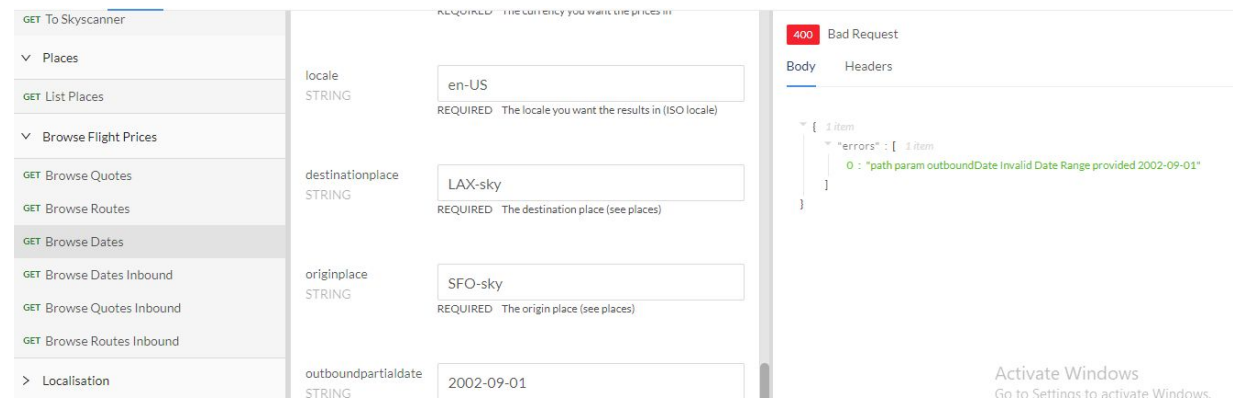


4. Select get Browse Routes 5. In country enter US 6. In currency enter USD 7. In locale enter en-US 8. In originplace enter SFO-sky 9. In destinationplace enter ORD-sky 10. In outboundpartialdate enter 2019-09-01 11. Click on Test Endpoint button	
<b>Expected result:</b>	
Getting error because date is incorrect	
<b>Actual result:</b>	
As expected	
<b>Pass/Fail:</b>	
Pass	
<b>Screenshot:</b>	
 <p>The screenshot shows a REST client interface with a sidebar on the left containing a list of endpoints under 'To Skyscanner'. The main area displays a GET request with the following parameters: locale (en-US), originplace (SFO-sky), destinationplace (ORD-sky), and outboundpartialdate (2019-09-01). The response is a 400 Bad Request with a JSON body containing an error message: 'path param outboundDate Invalid Date Range provided 2019-09-01'.</p>	

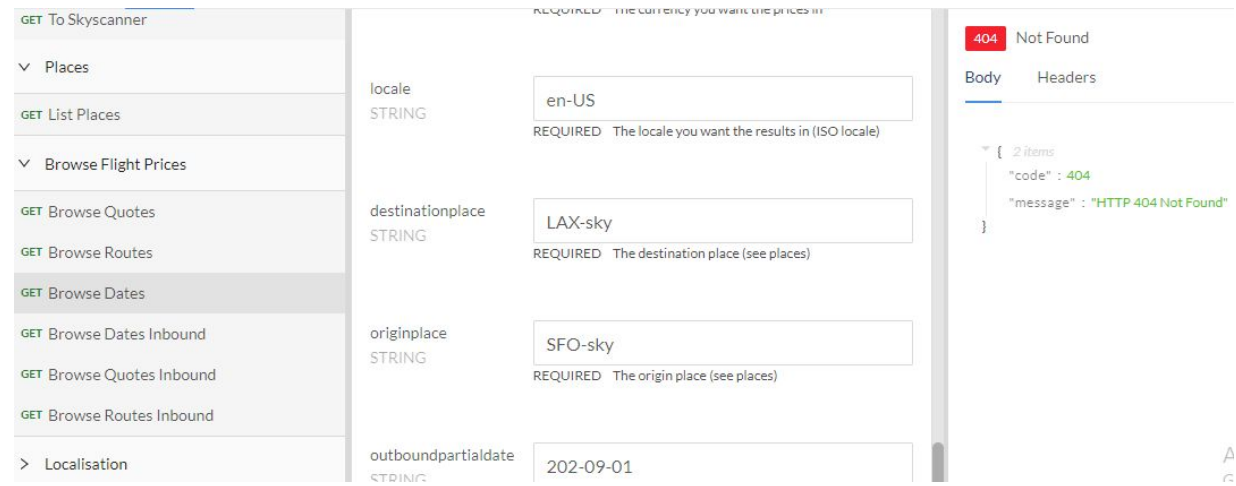
<b>Test Case: 009</b>	<b>Title:</b> Getting a list of browse dates
<b>Precondition:</b>	
User must be logged to skyscanner with Google, Facebook or Github account	
<b>Description:</b>	
Getting list of data including dates by sending valid data	
<b>Steps:</b>	
1. Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a>	

<ol style="list-style-type: none"> <li>Click on Test Endpoints button</li> <li>Click on Browse Flight Prices</li> <li>Select get Browse Dates</li> <li>In country enter US</li> <li>In currency enter USD</li> <li>In locale enter en-US</li> <li>In destinationplace enter LAX-sky</li> <li>In originplace enter SFO-sky</li> <li>In outboundpartialdate enter 2025-09-01</li> <li>Click on Test Endpoint button</li> </ol>	
<b>Expected result:</b>	
Getting list of data including expected dates	
<b>Actual result:</b>	
As expected	
<b>Pass/Fail:</b>	
Pass	
<b>Screenshot:</b>	
 <p>The screenshot displays the Skyscanner API test tool interface. On the left, a sidebar lists various endpoints, with 'GET Browse Dates' highlighted. The main area shows the configuration for this endpoint: 'country' is set to 'US', 'currency' to 'USD', 'locale' to 'en-US', and 'destinationplace' to 'LAX-sky'. Each field has a 'REQUIRED' note. On the right, the response is shown as a 200 OK status. The JSON body is expanded, showing a list of 5 items. The first item contains 'Quotes' (0 items), 'Carriers' (0 items), 'Places' (0 items), 'Currencies' (1 item), and 'Dates' (1 item). The 'Dates' item further shows 'OutboundDates' (0 items).</p>	

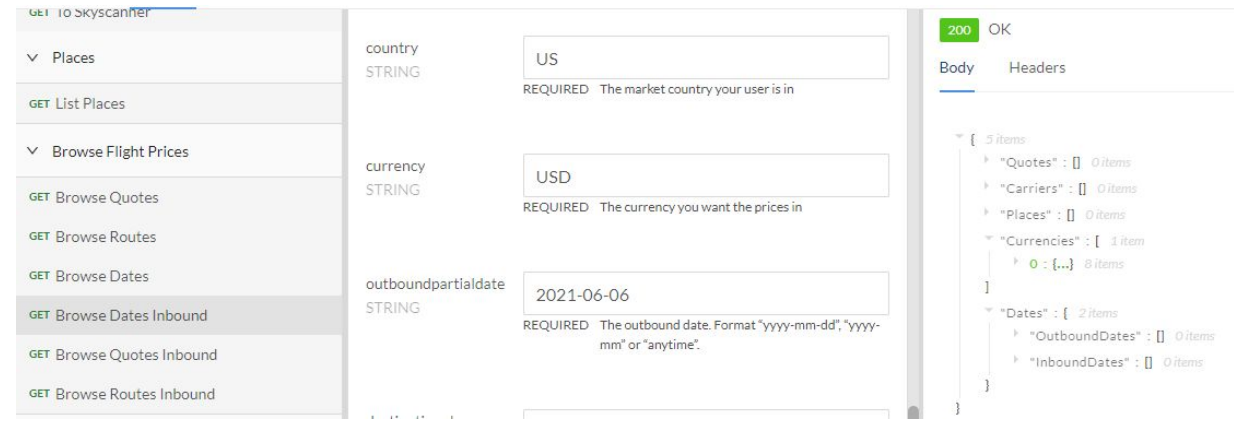
<b>Test Case: 010</b>	<b>Title:</b> Getting a list of browse dates
<b>Precondition:</b>	
User must be logged to skyscanner with Google, Facebook or Github account	
<b>Description:</b>	
Getting list of data including dates by sending invalid data	
<b>Steps:</b>	

<ol style="list-style-type: none"> <li>1. Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a></li> <li>2. Click on Test Endpoints button</li> <li>3. Click on Browse Flight Prices</li> <li>4. Select get Browse Dates</li> <li>5. In country enter US</li> <li>6. In currency enter USD</li> <li>7. In locale enter en-US</li> <li>8. In destinationplace enter LAX-sky</li> <li>9. In originplace enter SFO-sky</li> <li>10. In outboundpartialdate enter 2002-09-01</li> <li>11. Click on Test Endpoint button</li> </ol>	
<b>Expected result:</b>	
Getting error because date is incorrect	
<b>Actual result:</b>	
As expected	
<b>Pass/Fail:</b>	
Pass	
<b>Screenshot:</b>	
	

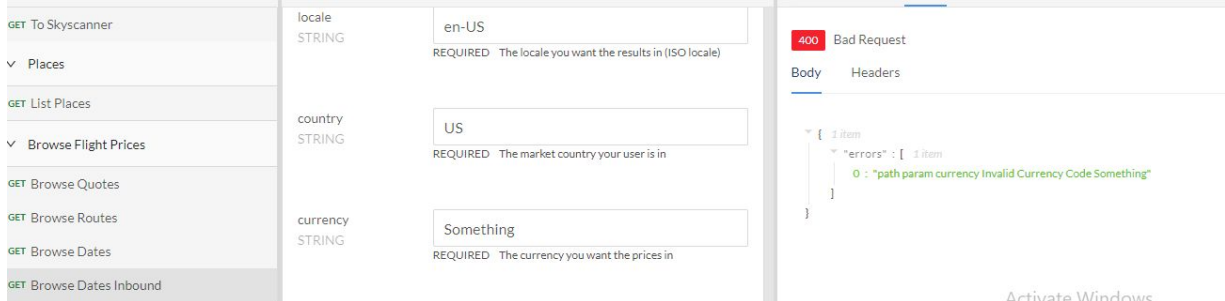
<b>Test Case: 011</b>	<b>Title:</b> Getting a list of browse dates
<b>Precondition:</b>	
User must be logged to skyscanner with Google, Facebook or Github account	
<b>Description:</b>	
Getting list of data including dates by sending invalid data	

<b>Steps:</b>	
<ol style="list-style-type: none"> <li>1. Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a></li> <li>2. Click on Test Endpoints button</li> <li>3. Click on Browse Flight Prices</li> <li>4. Select get Browse Dates</li> <li>5. In country enter US</li> <li>6. In currency enter USD</li> <li>7. In locale enter en-US</li> <li>8. In destinationplace enter LAX-sky</li> <li>9. In originplace enter SFO-sky</li> <li>10. In outboundpartialdate enter 202-09-01</li> <li>11. Click on Test Endpoint button</li> </ol>	
<b>Expected result:</b>	
Getting error because date is incorrect	
<b>Actual result:</b>	
As expected	
<b>Pass/Fail:</b>	
Pass	
<b>Screenshot:</b>	
	

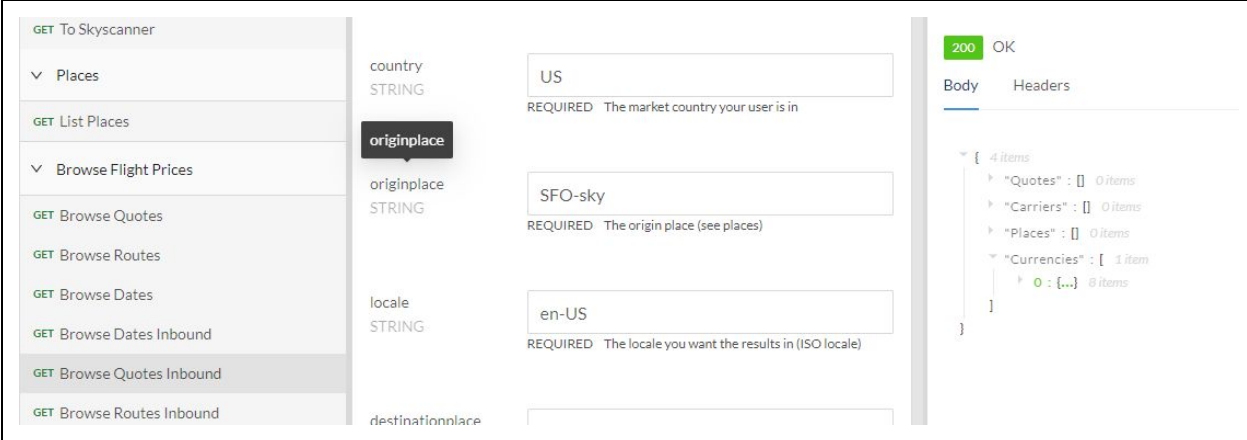
<b>Test Case: 012</b>	<b>Title:</b> Getting browse date inbound list
<b>Precondition:</b>	
User must be logged to skyscanner with Google, Facebook or Github account	

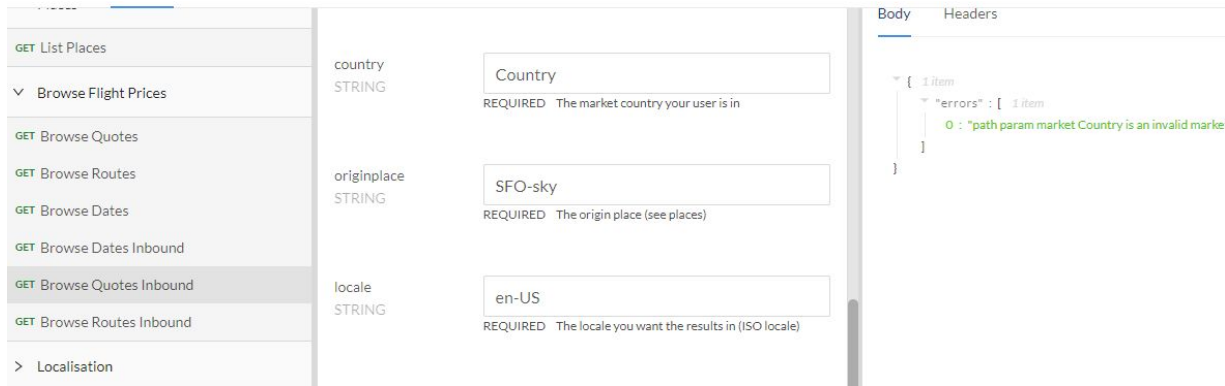
<b>Description:</b>	
Getting list of data including inbound date list by sending valid data	
<b>Steps:</b>	
<ol style="list-style-type: none"> <li>1. Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a></li> <li>2. Click on Test Endpoints button</li> <li>3. Click on Browse Flight Prices</li> <li>4. Select get Browse Dates Inbound</li> <li>5. In locale enter en-US</li> <li>6. In country enter US</li> <li>7. In currency enter USD</li> <li>8. In outboundpartialdate enter 2021-06-06</li> <li>9. In destinationplace enter JFK-sky</li> <li>10. In originplace enter SFO-sky</li> <li>11. In inboundpartialdate enter 2021-04-25</li> <li>12. Click on Test Endpoint button</li> </ol>	
<b>Expected result:</b>	
Getting list of data including expected inbound dates	
<b>Actual result:</b>	
As expected	
<b>Pass/Fail:</b>	
Pass	
<b>Screenshot:</b>	
 <p>The screenshot displays the RapidAPI interface for the Skyscanner API. On the left, a sidebar lists various endpoints, with 'GET Browse Dates Inbound' selected. The main area shows the configuration for this endpoint: 'country' is set to 'US', 'currency' to 'USD', and 'outboundpartialdate' to '2021-06-06'. A 'Test' button is visible at the bottom. On the right, the response is shown as a 200 OK status with a JSON body. The JSON structure includes 'Quotes', 'Carriers', 'Places', 'Currencies', and 'Dates'. The 'Dates' object contains 'OutboundDates' and 'InboundDates', both of which are currently empty arrays (0 items).</p>	

<b>Test Case: 013</b>	<b>Title: Getting a list of inbound data</b>
-----------------------	--

<b>Precondition:</b>	
User must be logged to skyscanner with Google, Facebook or Github account	
<b>Description:</b>	
Getting list of data including inbound dates by sending invalid data	
<b>Steps:</b>	
<ol style="list-style-type: none"> <li>1. Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a></li> <li>2. Click on Test Endpoints button</li> <li>3. Click on Browse Flight Prices</li> <li>4. Select get Browse Dates Inbound</li> <li>5. In locale enter en-US</li> <li>6. In country enter US</li> <li>7. In currency enter Something</li> <li>8. In outboundpartialdate enter 2021-06-06</li> <li>9. In destinationplace enter JFK-sky</li> <li>10. In originplace enter SFO-sky</li> <li>11. In inboundpartialdate enter 2021-04-25</li> <li>12. Click on Test Endpoint button</li> </ol>	
<b>Expected result:</b>	
Getting error because currency value is incorrect	
<b>Actual result:</b>	
As expected	
<b>Pass/Fail:</b>	
Pass	
<b>Screenshot:</b>	
	

<b>Test Case: 014</b>	<b>Title:</b> Getting a list of browse Quotes Inbound
-----------------------	---

<b>Precondition:</b>		
User must be logged to skyscanner with Google, Facebook or Github account		
<b>Description:</b>		
Getting list of data including routes by sending invalid data		
<b>Steps:</b>		
<ol style="list-style-type: none"> <li>1. Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a></li> <li>2. Click on Test Endpoints button</li> <li>3. Click on Browse Flight Prices</li> <li>4. Select get Browse Quotes Inbound</li> <li>5. In country enter US</li> <li>6. In currency enter USD</li> <li>7. In originplace enter SFO-sky</li> <li>8. In locale enter en-US</li> <li>9. In destinationplace enter ORD-sky</li> <li>10. In inboundpartialdate enter 2021-05-05</li> <li>11. In outboundpartialdate enter 2021-06-06</li> <li>12. Click on Test Endpoint button</li> </ol>		
<b>Expected result:</b>		
Getting list of data including expected quotes inbound list		
<b>Actual result:</b>		
As expected		
<b>Pass/Fail:</b>		
Pass		
<b>Screenshot:</b>		
 <p>The screenshot displays the RapidAPI interface for the Skyscanner API. On the left, a sidebar lists various endpoints, with 'Browse Flight Prices' selected. The main area shows the configuration for the 'GET Browse Quotes' endpoint. The input fields are as follows:</p> <ul style="list-style-type: none"> <li><b>country</b> (STRING): US (REQUIRED: The market country your user is in)</li> <li><b>originplace</b> (STRING): SFO-sky (REQUIRED: The origin place (see places))</li> <li><b>locale</b> (STRING): en-US (REQUIRED: The locale you want the results in (ISO locale))</li> <li><b>destinationplace</b>: (empty)</li> </ul> <p>The response status is 200 OK. The response body is shown as a JSON object with 4 items:</p> <pre> {   "Quotes": [],   "Carriers": [],   "Places": [],   "Currencies": [     {       "code": "USD",       "symbol": "\$",       "name": "US Dollar"     }   ] } </pre>		

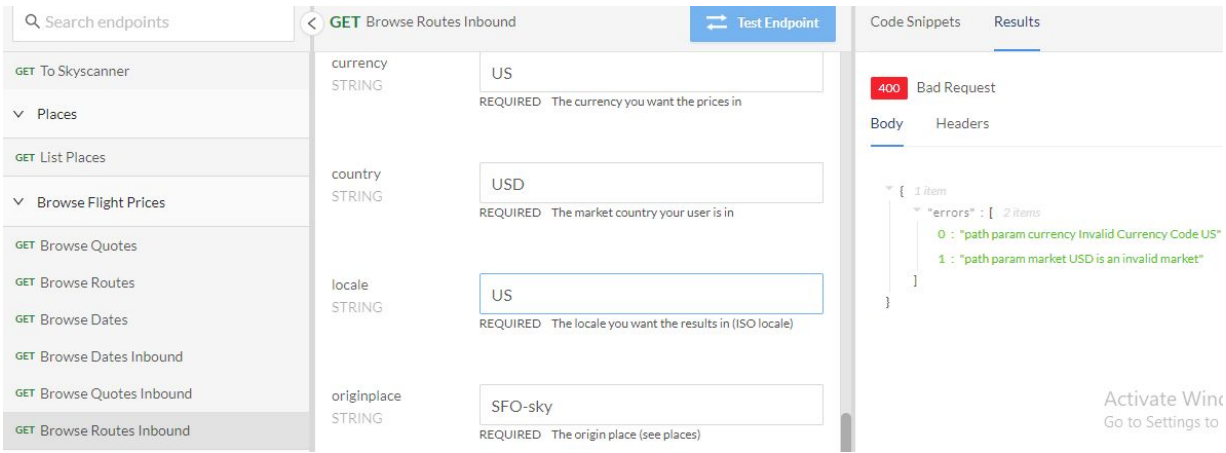
<b>Test Case: 015</b>	<b>Title:</b> Getting a list of browse quotes inbound
<b>Precondition:</b>	
User must be logged to skyscanner with Google, Facebook or Github account	
<b>Description:</b>	
Getting list of data including quotes by sending invalid data	
<b>Steps:</b>	
<ol style="list-style-type: none"> <li>1. Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a></li> <li>2. Click on Test Endpoints button</li> <li>3. Click on Browse Flight Prices</li> <li>4. Select get Browse Quotes Inbound</li> <li>5. In country enter Country</li> <li>6. In currency enter USD</li> <li>7. In originplace enter SFO-sky</li> <li>8. In locale enter en-US</li> <li>9. In destinationplace enter ORD-sky</li> <li>10. In inboundpartialdate enter 2021-05-05</li> <li>11. In outboundpartialdate enter 2021-06-06</li> <li>13. Click on Test Endpoint button</li> </ol>	
<b>Expected result:</b>	
Getting error because country value is incorrect	
<b>Actual result:</b>	
As expected	
<b>Pass/Fail:</b>	
Pass	
<b>Screenshot:</b>	
 <p>The screenshot displays the RapidAPI interface for the Skyscanner API. On the left, a sidebar lists various endpoints, with 'GET Browse Quotes Inbound' selected. The main area shows the configuration for this endpoint, including input fields for 'country' (set to 'Country'), 'originplace' (set to 'SFO-sky'), and 'locale' (set to 'en-US'). Each field has a 'REQUIRED' status and a descriptive note. On the right, the 'Body' tab shows the JSON response, which contains an 'errors' array with a single message: 'path param market Country is an invalid market'.</p>	



<b>Test Case: 016</b>	<b>Title:</b> Getting a list of browse routes inbound
<b>Precondition:</b>	
User must be logged to skyscanner with Google, Facebook or Github account	
<b>Description:</b>	
Getting list of data including routes by sending valid data	
<b>Steps:</b>	
<ol style="list-style-type: none"> <li>1. Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a></li> <li>2. Click on Test Endpoints button</li> <li>3. Click on Browse Flight Prices</li> <li>4. Select get Browse Routes Inbound</li> <li>5. In country enter US</li> <li>6. In currency enter USD</li> <li>7. In locale enter en-US</li> <li>8. In originplace enter SFO-sky</li> <li>9. In outboundpartialdate enter 2022-03-03</li> <li>10. In inboundpartial date enter 2021-03-03</li> <li>11. In destinationplace enter ORD-sky</li> <li>12. Click on Test Endpoint button</li> </ol>	
<b>Expected result:</b>	
Getting list of data including expected routes	
<b>Actual result:</b>	
As expected	
<b>Pass/Fail:</b>	
Pass	
<b>Screenshot:</b>	

GET To Skyscanner	currency STRING	USD <small>REQUIRED The currency you want the prices in</small>	<div>200 OK</div> <div>Body Headers</div> <pre> {   "Quotes": [],   "Carriers": [],   "Places": [],   "Currencies": [     {       "code": "USD",       "symbol": "\$",       "name": "US Dollar"     }   ],   "Routes": [] } </pre>
Places			
GET List Places	country STRING	US <small>REQUIRED The market country your user is in</small>	
Browse Flight Prices			
GET Browse Quotes	locale STRING	en-US <small>REQUIRED The locale you want the results in (ISO locale)</small>	
GET Browse Routes	originplace STRING	SFO-sky <small>REQUIRED The origin place (see places)</small>	
GET Browse Dates			
GET Browse Dates Inbound			
GET Browse Quotes Inbound			
GET Browse Routes Inbound	outboundpartialdate STRING	2022-03-03	
Localisation			

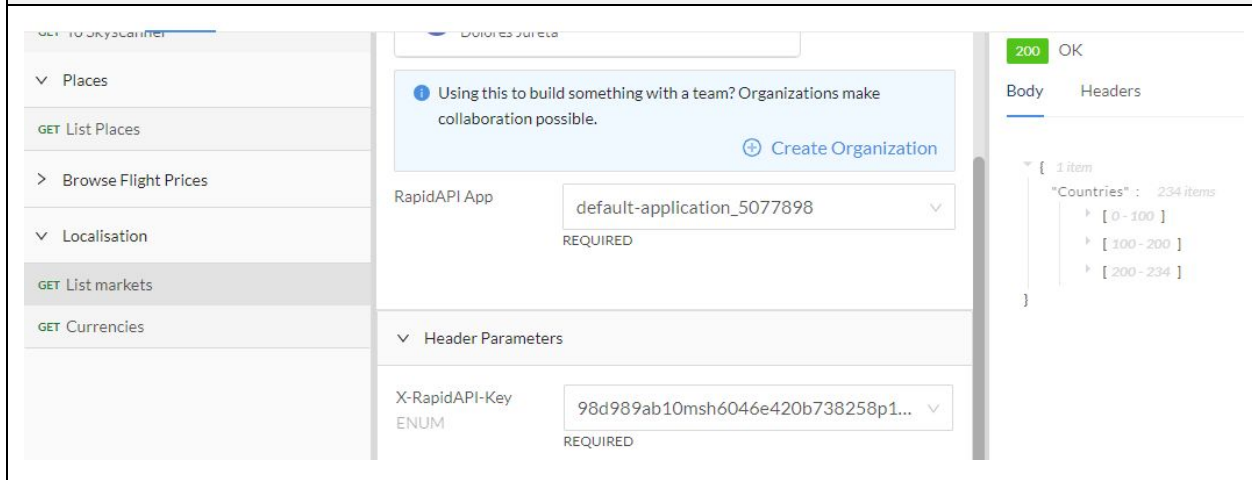
<b>Test Case: 017</b>	<b>Title: Getting a list of browse routes inbound</b>
<b>Precondition:</b>	
User must be logged to skyscanner with Google, Facebook or Github account	
<b>Description:</b>	
Getting list of data including routes by sending invalid data	
<b>Steps:</b>	
<ol style="list-style-type: none"> <li>Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a></li> <li>Click on Test Endpoints button</li> <li>Click on Browse Flight Prices</li> <li>Select get Browse Routes Inbound</li> <li>In country enter USD</li> <li>In currency enter US</li> <li>In locale enter US</li> <li>In originplace enter SFO-sky</li> <li>In outboundpartialdate enter 2022-03-03</li> <li>In inboundpartial date enter 2021-03-03</li> <li>In destinationplace enter ORD-sky</li> <li>Click on Test Endpoint button</li> </ol>	
<b>Expected result:</b>	
Getting error because values country and currency are incorrect	
<b>Actual result:</b>	
As expected	

<b>Pass/Fail:</b>	
Pass	
<b>Screenshot:</b>	
	

<b>Test Case: 018</b>	<b>Title:</b> Getting a list of markets
<b>Precondition:</b>	
User must be logged to skyscanner with Google, Facebook or Github account	
<b>Description:</b>	
Getting list of markets by sending valid data	
<b>Steps:</b>	
<ol style="list-style-type: none"> <li>1. Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a></li> <li>2. Click on Test Endpoints button</li> <li>3. Click on Localization</li> <li>4. Select get List Markets</li> <li>5. In locale enter en-US</li> <li>6. Click on Test Endpoint button</li> </ol>	
<b>Expected result:</b>	
Getting list of markets	
<b>Actual result:</b>	
As expected	
<b>Pass/Fail:</b>	

Pass

### Screenshot:



**Test Case: 019**

**Title:** Getting a list of markets

### Precondition:

User must be logged to skyscanner with Google, Facebook or Github account

### Description:

Getting list of markets by sending invalid data

### Steps:

1. Go to site <https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details>
2. Click on Test Endpoints button
3. Click on Localization
4. Select get List Markets
5. In locale enter en-US
6. Click on Test Endpoint button

### Expected result:

Getting error because local value is incorrect

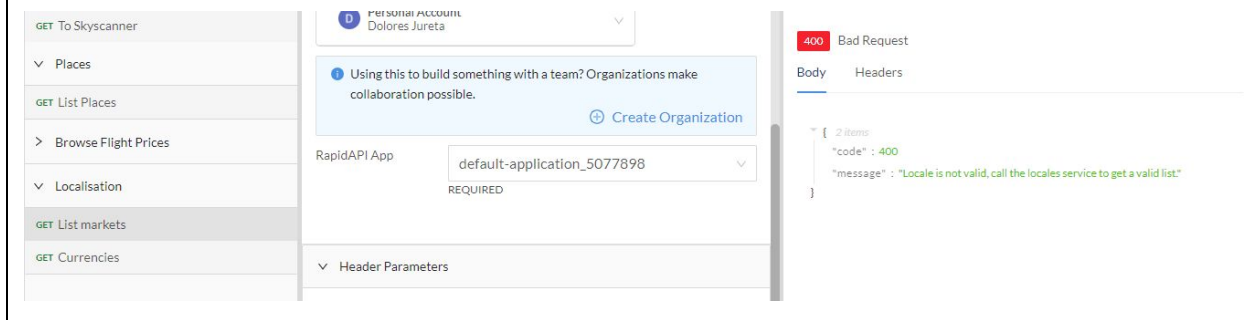
### Actual result:

As expected

### Pass/Fail:

Pass

## Screenshot:



**Test Case: 020**

**Title:** Getting a list of currencies

### Precondition:

User must be logged to skyscanner with Google, Facebook or Github account

### Description:

Getting list of currency by sending valid data

### Steps:

1. Go to site <https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details>
2. Click on Test Endpoints button
3. Click on Localization
4. Select get List Currency
5. Click on Test Endpoint button

### Expected result:

Getting list of currencies

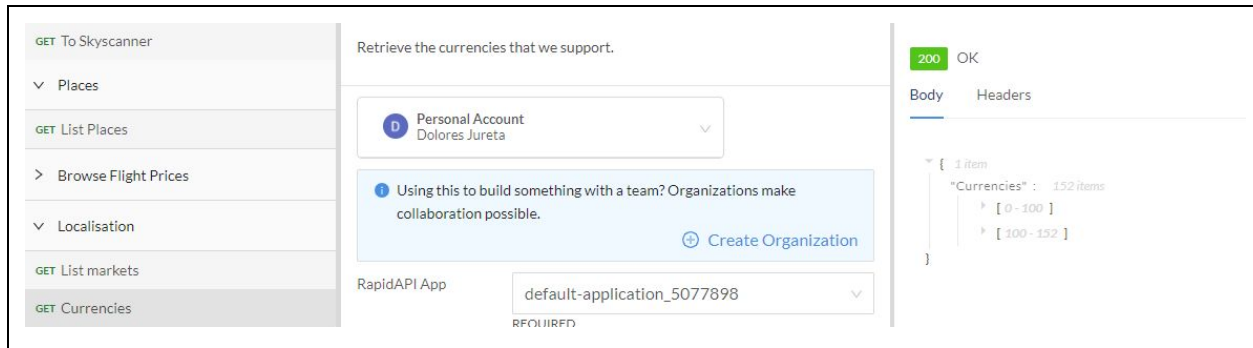
### Actual result:

As expected

### Pass/Fail:

Pass

### Screenshot:



Test Case: 021	Title: Getting flights on Skyscanner
<b>Precondition:</b>	
User must be logged to skyscanner with Google, Facebook or Github account	
<b>Description:</b>	
Getting flight offers by sending valid data	
<b>Steps:</b>	
<ol style="list-style-type: none"> <li>1. Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a></li> <li>2. Click on Test Endpoints button</li> <li>3. Click on get to Skyscanner</li> <li>4. In country enter US</li> <li>5. In currency enter USD</li> <li>6. In locale enter en-US</li> <li>7. In originplace enter SFO-sky</li> <li>8. In destinationplace enter ORD-sky</li> <li>9. In outboundpartialdate enter 2021-08-08</li> <li>10. Click on Test Endpoint button</li> </ol>	
<b>Expected result:</b>	
Getting flight offers	
<b>Actual result:</b>	
Getting error that service is not available	
<b>Pass/Fail:</b>	
Fail	
<b>Screenshot:</b>	

## 2. Identifying test cases that could represent a smoke test

Definition: Smoke tests are a subset of test cases that cover the most important functionality of a component or system.

In this application test cases that could represent smoke test are 001, 006, 018, 020, because they represent basic flow that should be available to users when finding cheapest routes.

## 3. Identifying negative and positive test cases

	Number of test case
<b>Positive test cases</b>	001, 003, 006, 009, 012, 016, 018, 020, 021
<b>Negative test cases</b>	002, 004, 005, 007, 008, 010, 011, 013, 014, 015, 017, 019

## 6. Reporting bugs

<b>Bug: 001</b>	<b>Related Test Case: 002</b>
<b>Title:</b>	
Sending invalid values for string query	
<b>Description:</b>	
Inserting only one char in field that requires at least two chars	
<b>Steps to reproduce</b>	
<ul style="list-style-type: none"> <li>Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a></li> <li>Click on Test Endpoints button</li> <li>Click on Places</li> <li>Select get List Places</li> <li>In query enter B</li> <li>In country enter UK</li> <li>In currency enter GBP</li> </ul>	

<ul style="list-style-type: none"> <li>• In locale enter en-GB</li> <li>• Click on Test Endpoint button</li> </ul>
<b>Expected result:</b>
Fail: String query must be at least 2 characters
<b>Actual result:</b>
Pass (Code: 200)
<b>Severity:</b>
Low-medium

<b>Bug: 002</b>	<b>Related Test Case: 021</b>
<b>Title:</b>	
Getting flight offers on skyscanner	
<b>Description:</b>	
Inserting valid values and trying to get flight offers	
<b>Steps to reproduce</b>	
<ul style="list-style-type: none"> <li>• Go to site <a href="https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details">https://rapidapi.com/skyscanner/api/skyscanner-flight-search/details</a></li> <li>• Click on Test Endpoints button</li> <li>• Click on get to Skyscanner</li> <li>• In country enter US</li> <li>• In currency enter USD</li> <li>• In locale enter en-US</li> <li>• In originplace enter SFO-sky</li> <li>• In destinationplace enter ORD-sky</li> <li>• In outboundpartialdate enter 2021-08-08</li> <li>• Click on Test Endpoint button</li> </ul>	
<b>Expected result:</b>	
Fail: The service is not available	
<b>Actual result:</b>	
Fail (Code: 404)	
<b>Severity:</b>	
Critical	