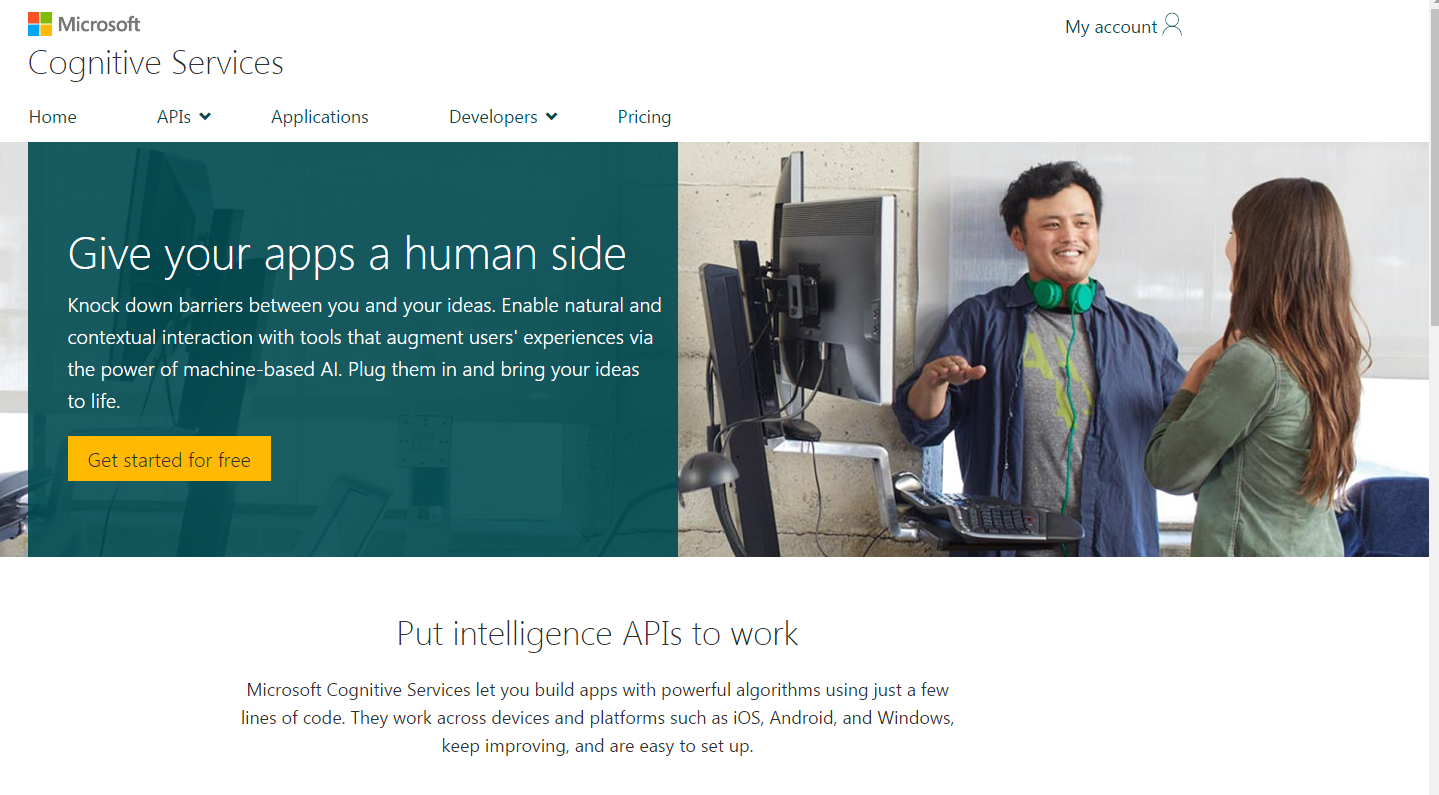
# Introduction

Microsoft Cognitive Services is a set of APIs that use the power of Machine Learning to analyze, pictures, video, speech, and language.

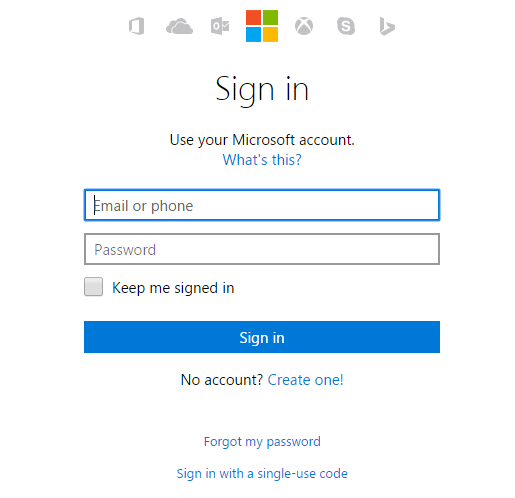
Working through this set of exercises will help you understand how to call these APIs, what data is required, and what data is returned.

# Exercise 1: Sign up and Explore Cognitive Services

Open a browser and navigate to the Cognitive Services home page at <https://www.microsoft.com>.

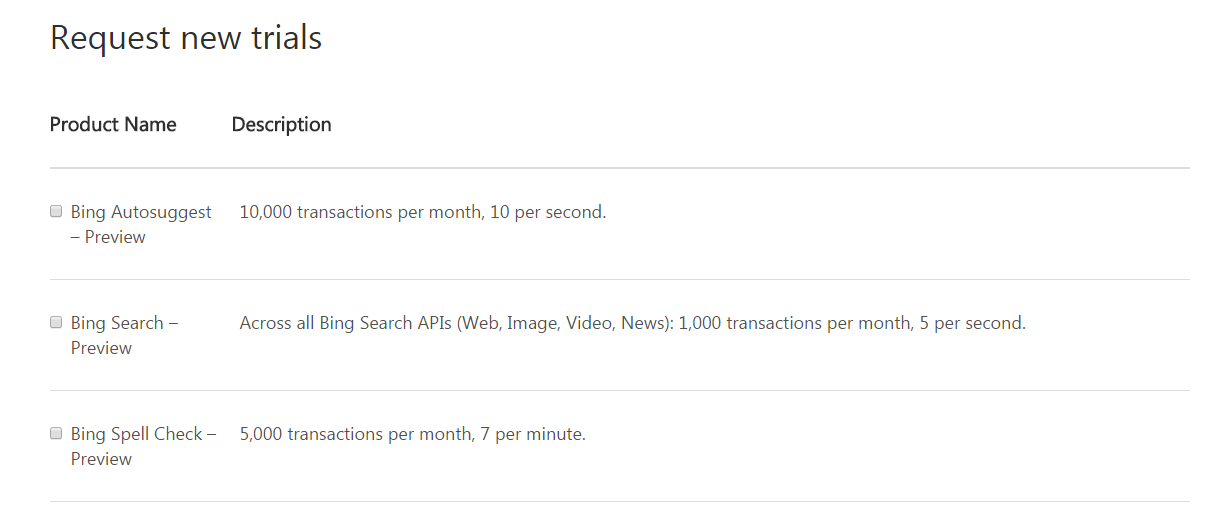


At the top left, click the “My Account” link.

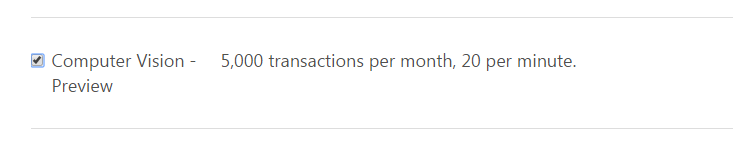


If you are not yet signed in, you will need to sign in with a “Microsoft account”, which can be an outlook.com account, a hotmail.com account, a live.com account, or any email account that you have associated as your sign in form Microsoft services. If you have no such account, the “Create One” link allows you to create one. Sign in with your Microsoft account to return to the “My Account” page.

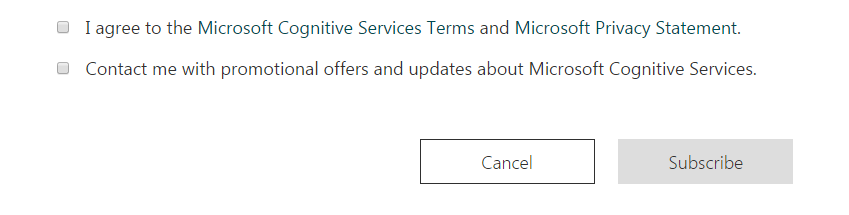
If this is the first time using Cognitive Services, you will be presented with a list of APIs.



You can check the checkbox next to any API you wish to use. In the exercises below, you will be using the Computer Vision API. Scroll down and check the checkbox next to “Computer Vision”.

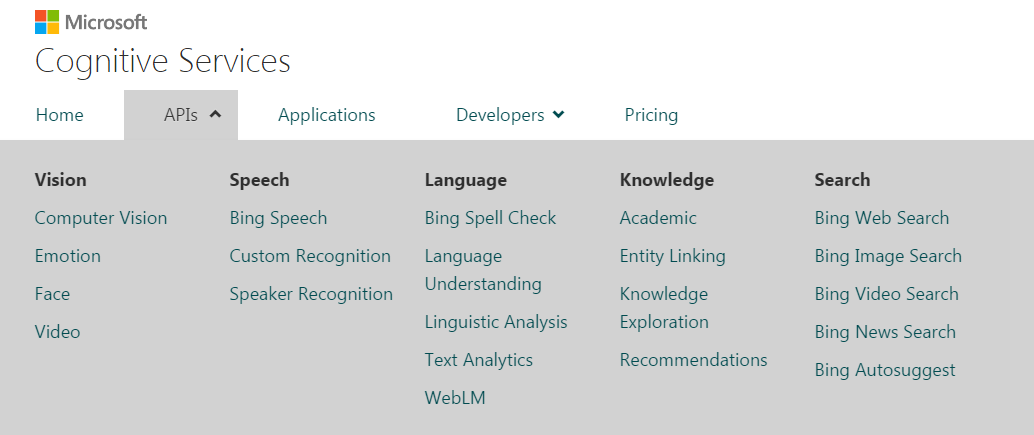


Scroll to the bottom of the page and check the checkbox next to “I agree to the Microsoft Cognitive Services Terms and Microsoft Privacy Statement”; then, click the “Subscribe” button.

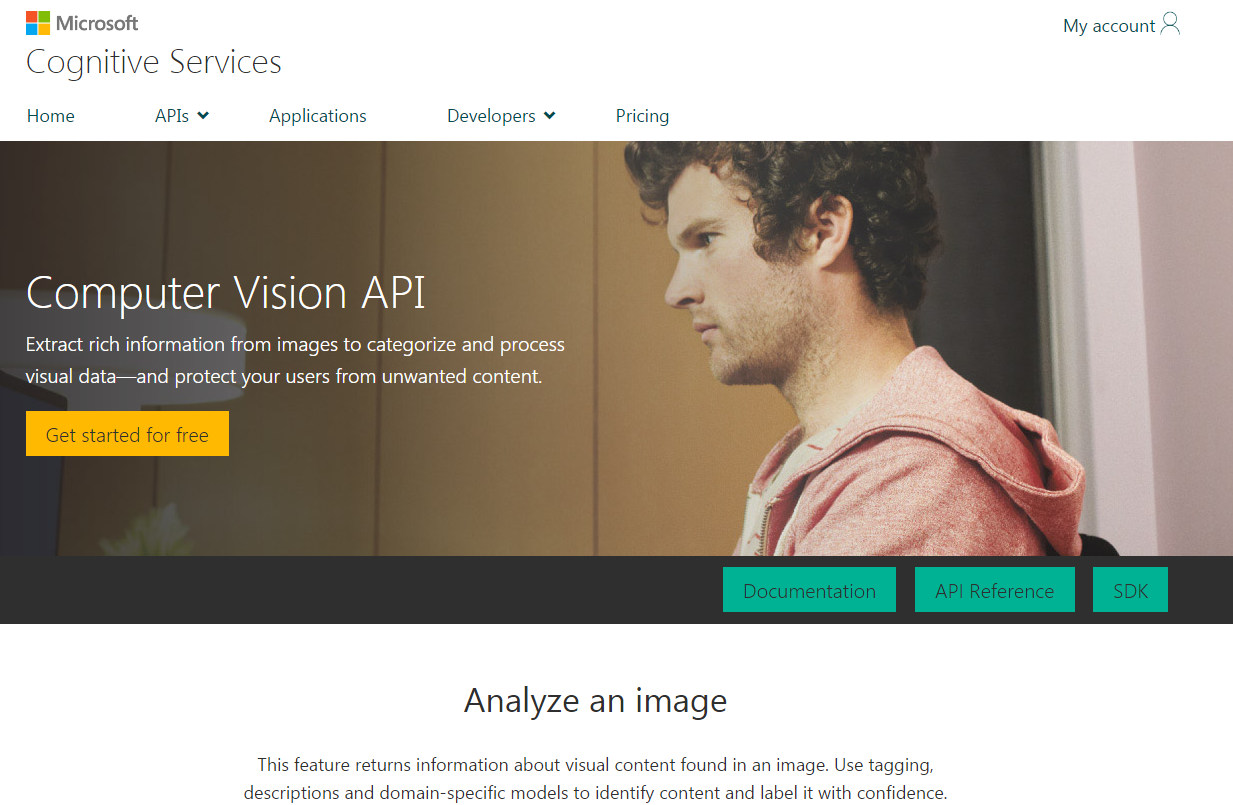


# Exercise 2: Use the sample application

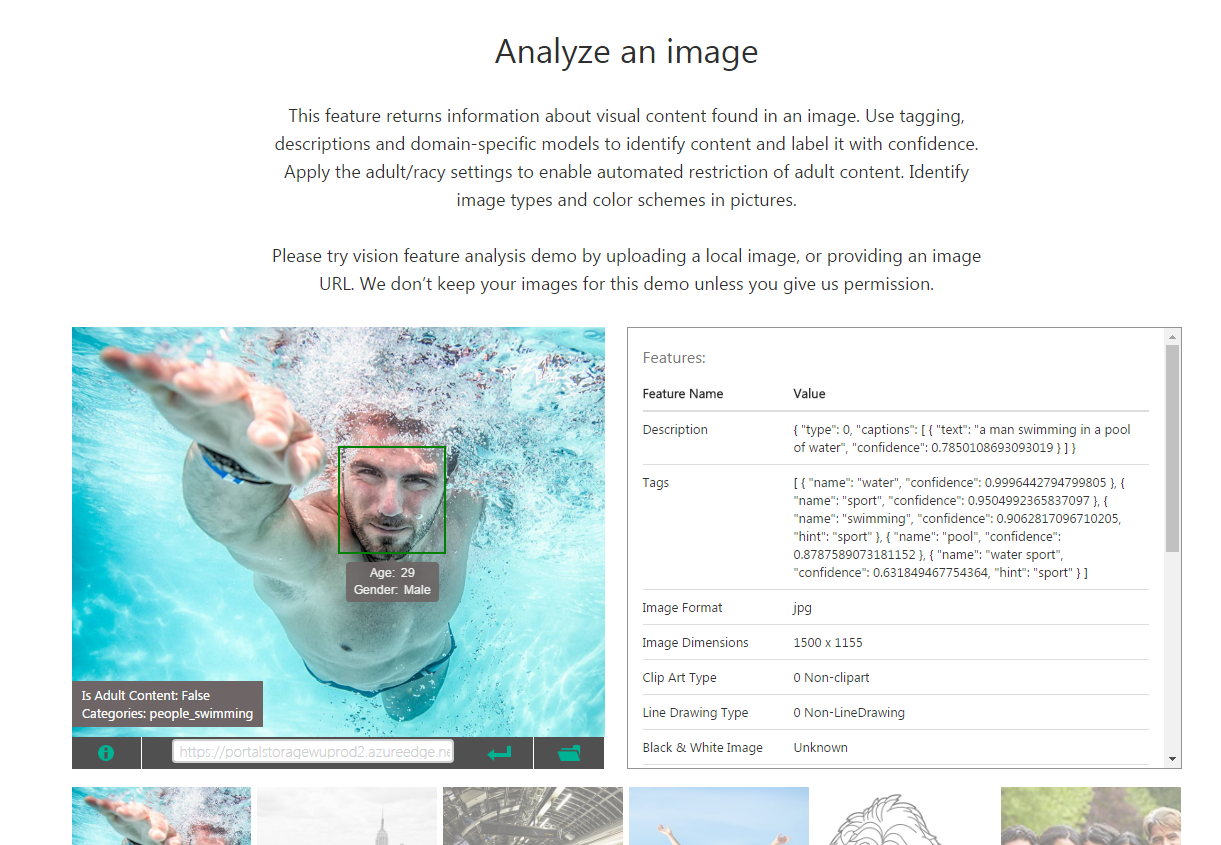
Click the “APIs” link near the top of the page to expand the APIs menu.



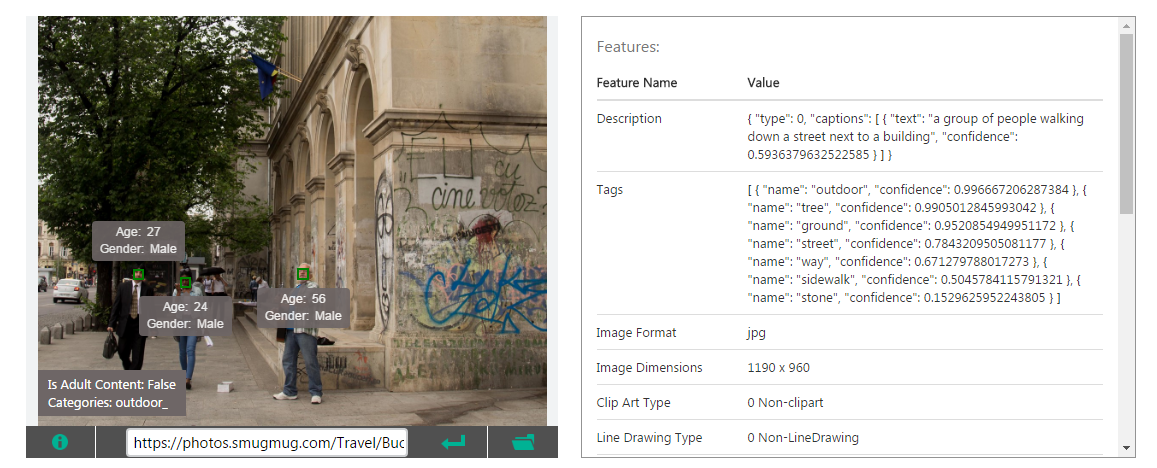
Under the “Vision” column, select “Computer Vision” to display the “Computer Vision API” page.



Scroll down to the “Analyze an Image” section of the “Computer Vision API” page.



Select a picture from among the thumbnails. A large version of the selected picture will display in the frame above. After a few seconds, you should see the Features of that picture displayed to the left of the large picture. Examine these features to determine if they accurately describe the picture.

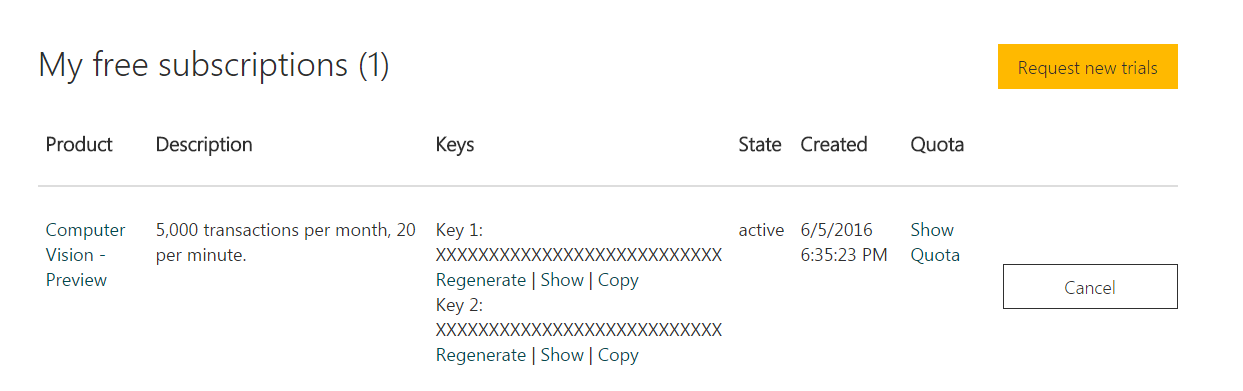
Find the URL of an interesting picture on the web (use <http://www.bing.com/images> to help you find one). Paste that URL in the textbox below the large picture and click the “Submit” button (). After a few seconds, you should see the Features of that picture displayed to the left of the large picture. Examine these features to determine if they accurately describe the picture. If labels do not appear on the picture, click the “Show Information” button () to display them.

# Exercise 3: Call a Web Service

Most Cognitive Services APIs include a web form to call the API from within a browser and view the HTTP request and response of that call. In this exercise, you will use this tool to call the Computer Vision API.

For this exercise, you will need the API key for the Computer Vision API. Scroll down to the “Computer Vision”.

Click the “My Account” to navigate to the Account Subscriptions page or navigate to <https://www.microsoft.com/cognitive-services/en-US/subscriptions>.

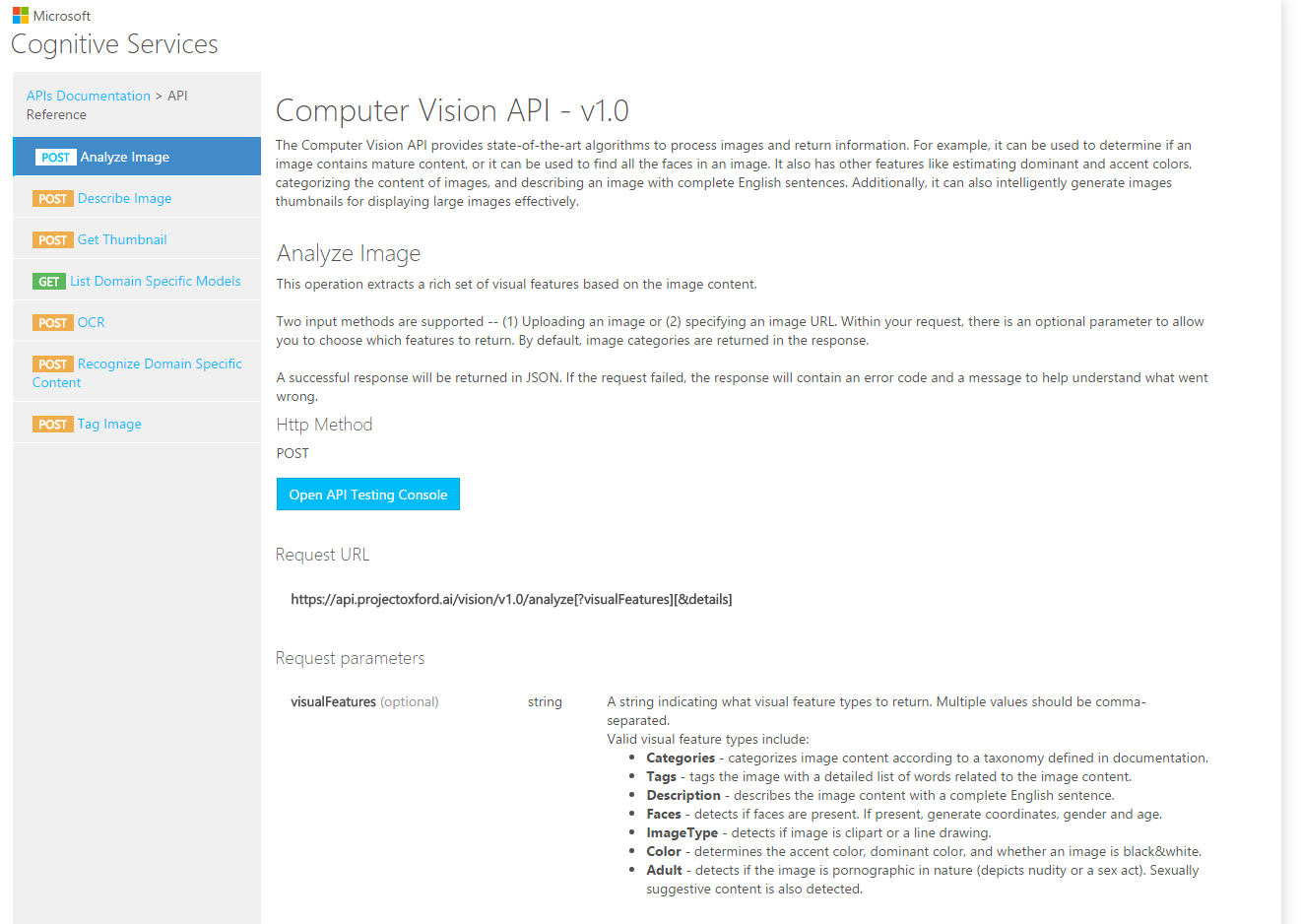


The 2 API keys for each API are obscured by X’s. Next to “Computer Vision”, click the “Show” link below Key 1 to reveal the API Key. Click the “Copy” link to copy this to your clipboard. Save this somewhere for later in this exercise or return to this page when you need the API key.

Navigate to the “Computer Vision API” page (<https://www.microsoft.com/cognitive-services/en-us/computer-vision-api>) and click the “API Reference” button.



A page opens, listing all the functionality available in the Computer Vision API. Click the “Analyze Image” button in the left sidebar to display information on this API.

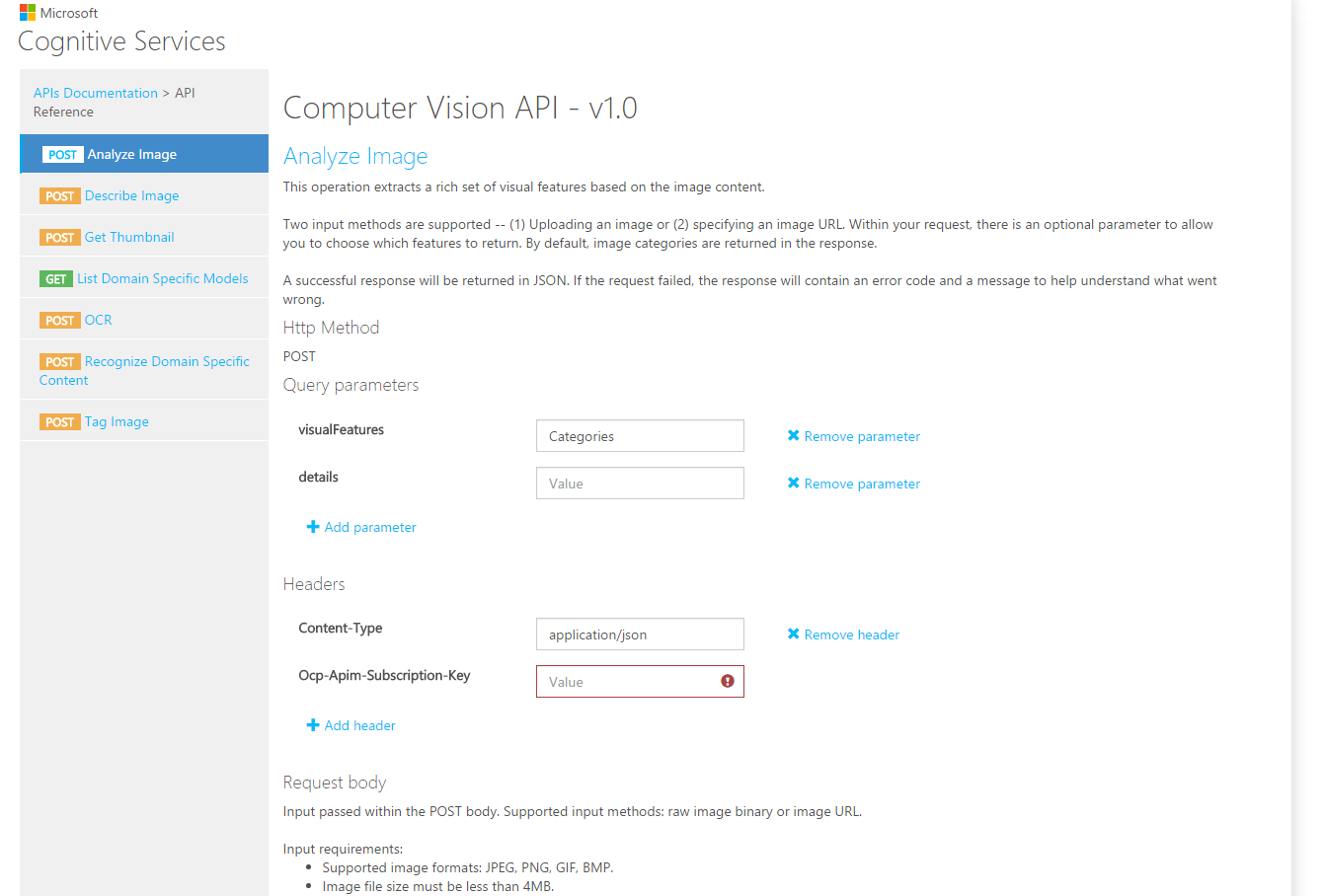


Scroll down the page to see an explanation of how to call this service. Basically, you must send a POST HTTP request to a well-known URL and pass to the URL some information (the content type and subscription key) in the HTTP header; and some information as querystring parameters. Each parameter is explained and sample data is shown, along with possible responses and sample JSON data returned by this service.

Click the “Open API Testing Console” button near the top of the page.



The API Testing Console page allows you to enter or select parameters and call the API from within your browser.

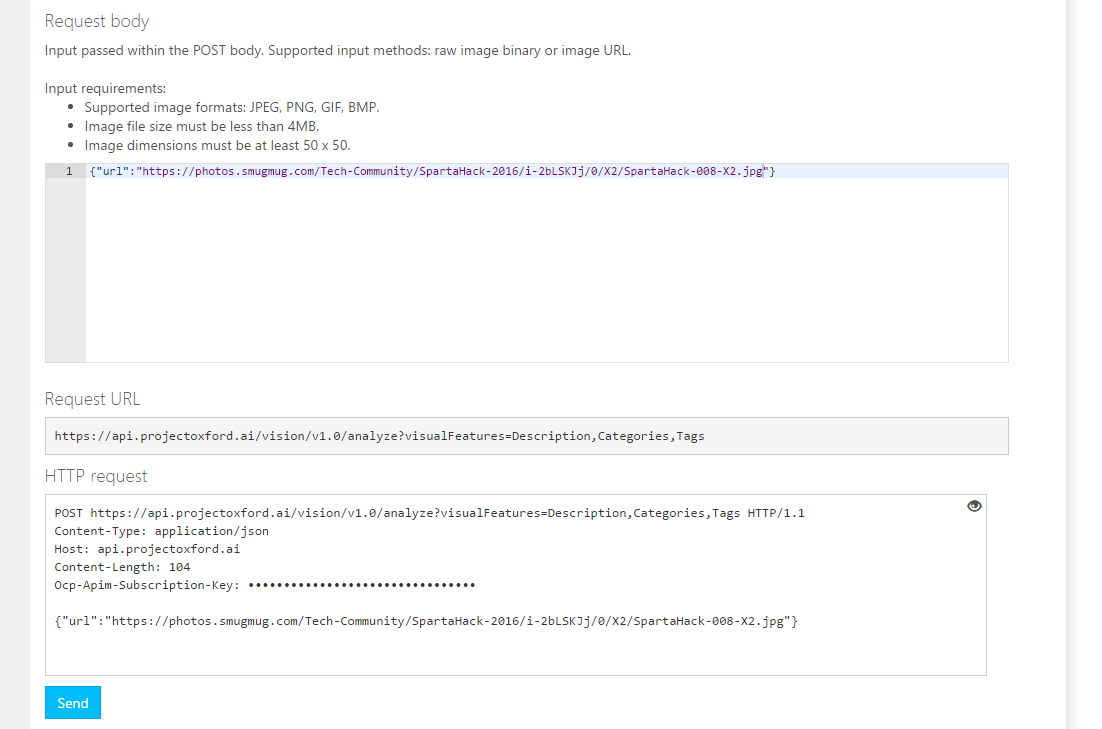


At the “visualFeatures” text box, enter “Description,Categories,Tags”.

Click the “Remove parameter” link () next to the “details” text box to remove this row.

At the “Content-Type” text box, select or type “application/json”.

At the “Ocp-Apim-Subscription-Key” text box, paste in your Computer Vision API key that you found at the start of this exercise.

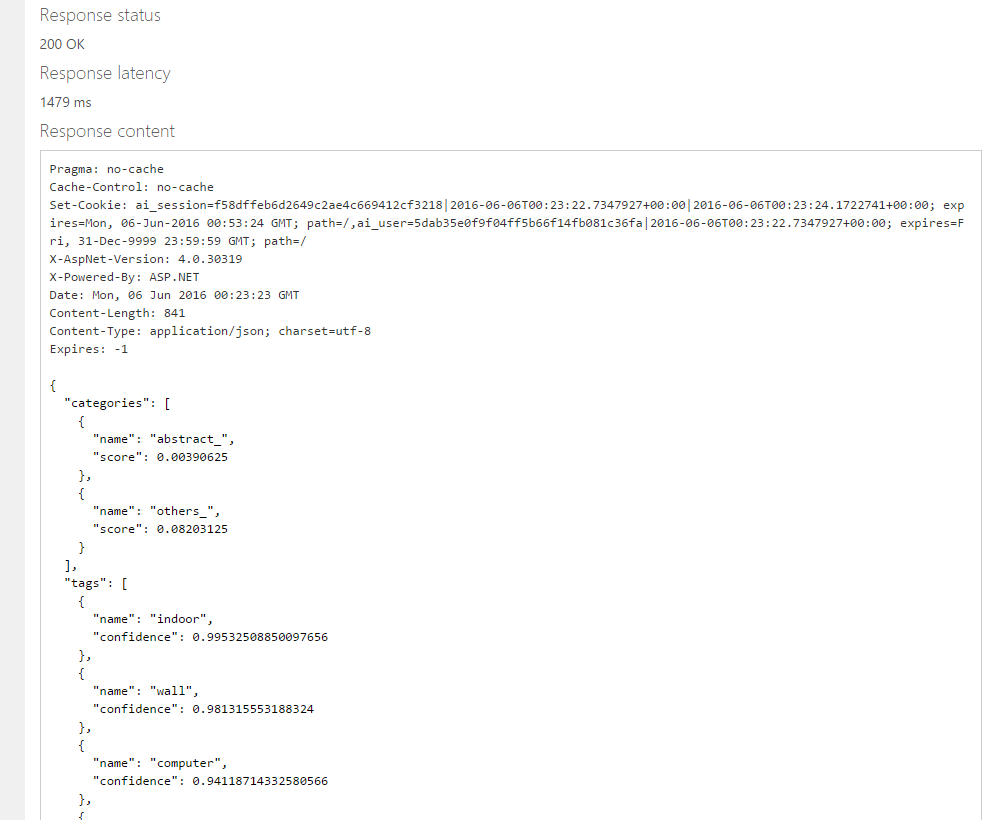


The “Request Body” text area contains an example of JSON to show you the format in which to enter the data. Replace the URL with an actual URL pointing to a picture on the web (you can use <http://www.bing.com/images> to search for a picture or just use <https://photos.smugmug.com/Tech-Community/SpartaHack-2016/i-2bLSKJj/0/X2/SpartaHack-008-X2.jpg>).

As you type, notice how the “Request URL” and “HTTP Request” sections automatically update to reflect the request that will be sent to this web service.

Click the “Send” button to initiate a call to the web service.

Scroll down to see the HTTP Response and JSON returned.



If the call was successful, the “Response status” will be “200 OK” and the “Response content” will contain JSON with information about the picture. Review this information to see how accurate the service interpreted the contents of the picture. Is the caption accurate? Are the tags and categories appropriate?

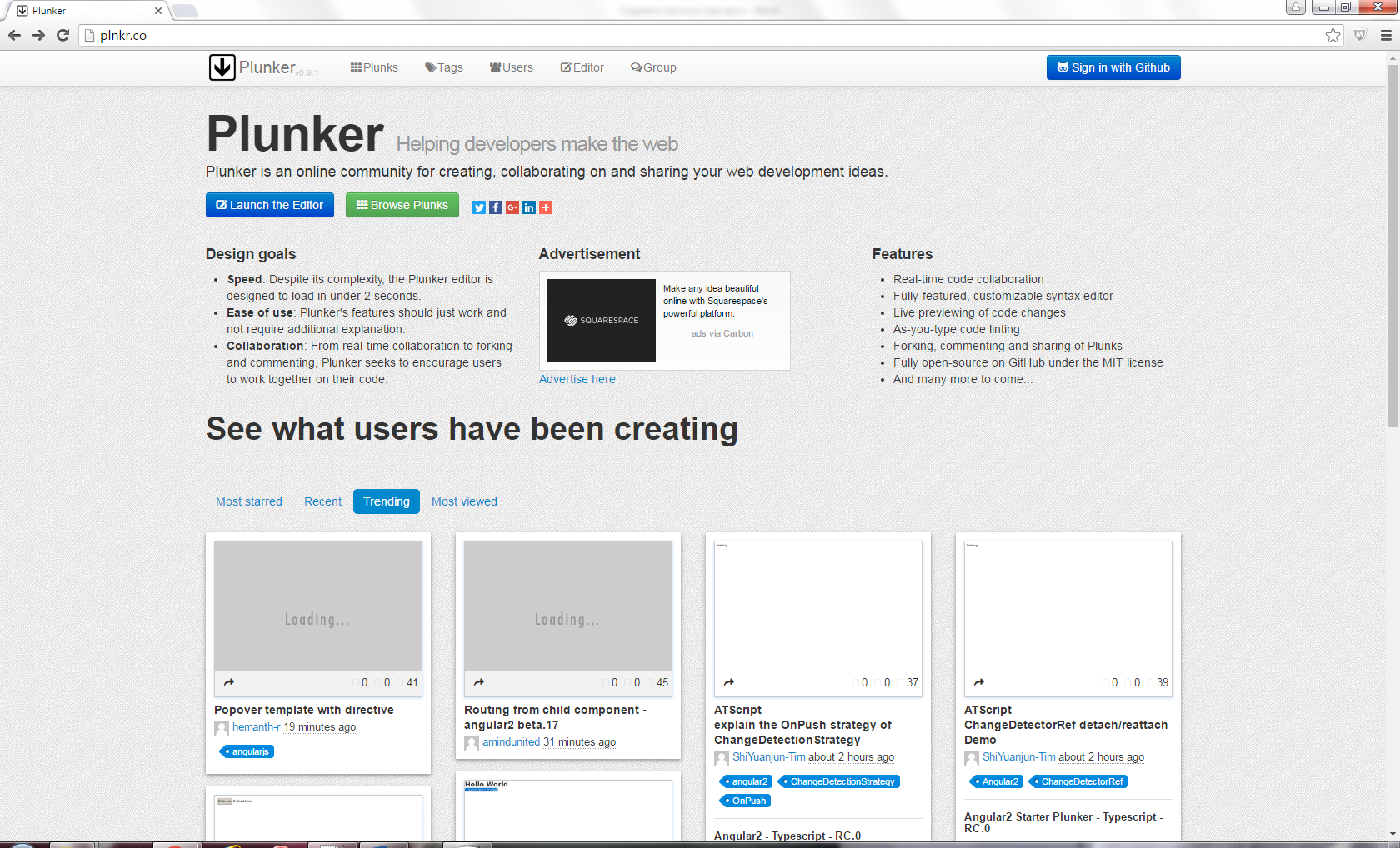
If the call was unsuccessful, the response status will be something other than “200 OK”. A common error is “401 Access Denied”, which often means that you copied an incorrect Subscription Key.

# Exercise 4: Calling a Web Service from JavaScript

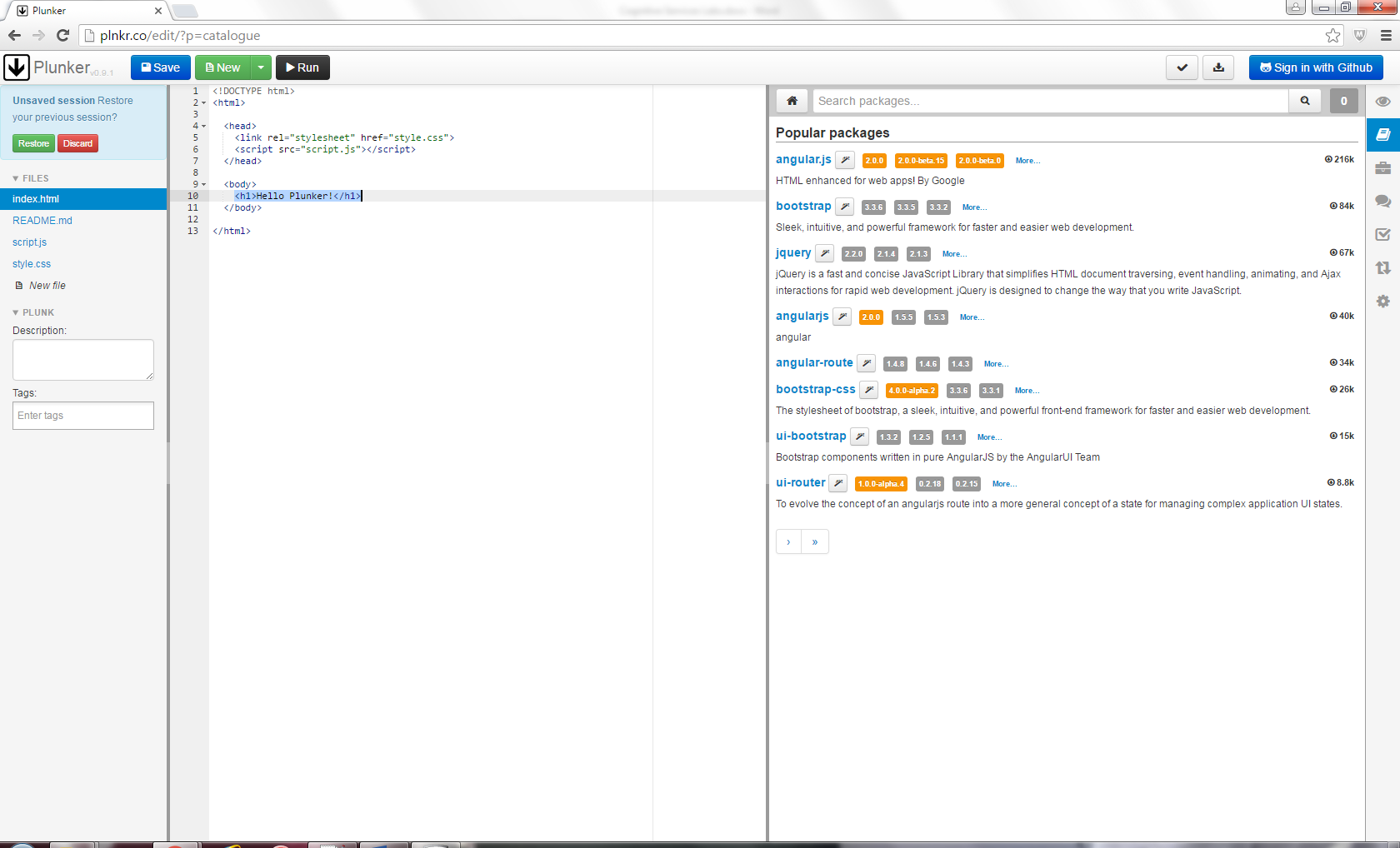
In this exercise, we will write some JavaScript to call the Image Analysis API and examine the data returned.

We will use Plunker – a browser-based tool for writing and running JavaScript, so that you will not need to install anything on your local machine.

Open a browser and navigate to <http://plnkr.co/>.



Click the “Launch the Editor” button () to (you guessed it!) launch the editor.



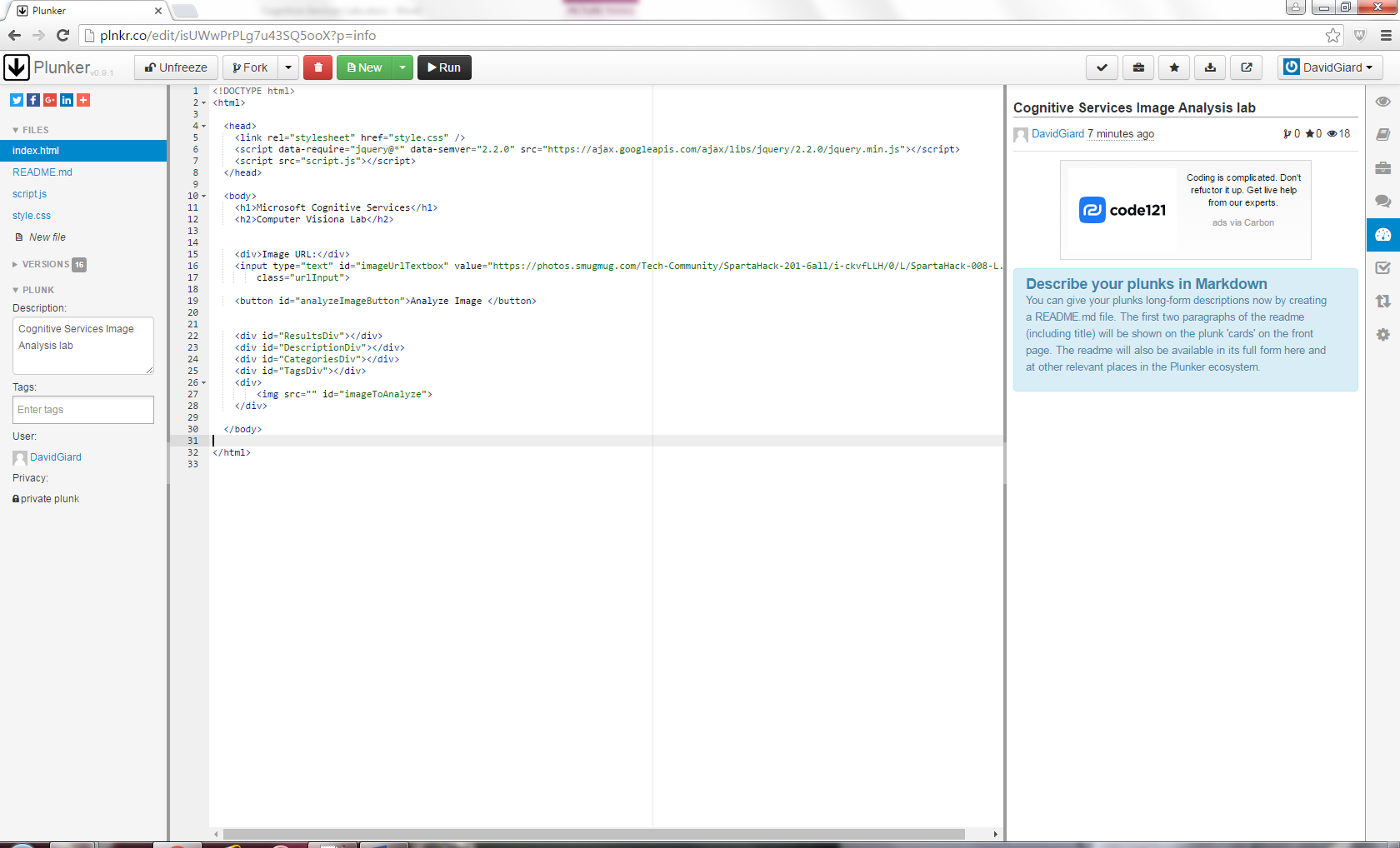
In the “Popular packages” pane, click the most recent version of “jquery”.



Replace the HTML “<h1>Hello Plunkr</h1> with the following:

|  |
| --- |
| <h1>Microsoft Cognitive Services</h1>  <h2>Computer Visiona Lab</h2>  <div>Image URL:</div>  <input type="text" id="imageUrlTextbox" value="https://photos.smugmug.com/Tech-Community/SpartaHack-201-6all/i-ckvfLLH/0/L/SpartaHack-008-L.jpg"  class="urlInput">  <button id="analyzeImageButton">Analyze Image </button>  <div id="ResultsDiv"></div>  <div id="DescriptionDiv"></div>  <div id="CategoriesDiv"></div>  <div id="TagsDiv"></div>  <div>  <img src="" id="imageToAnalyze">  </div> |

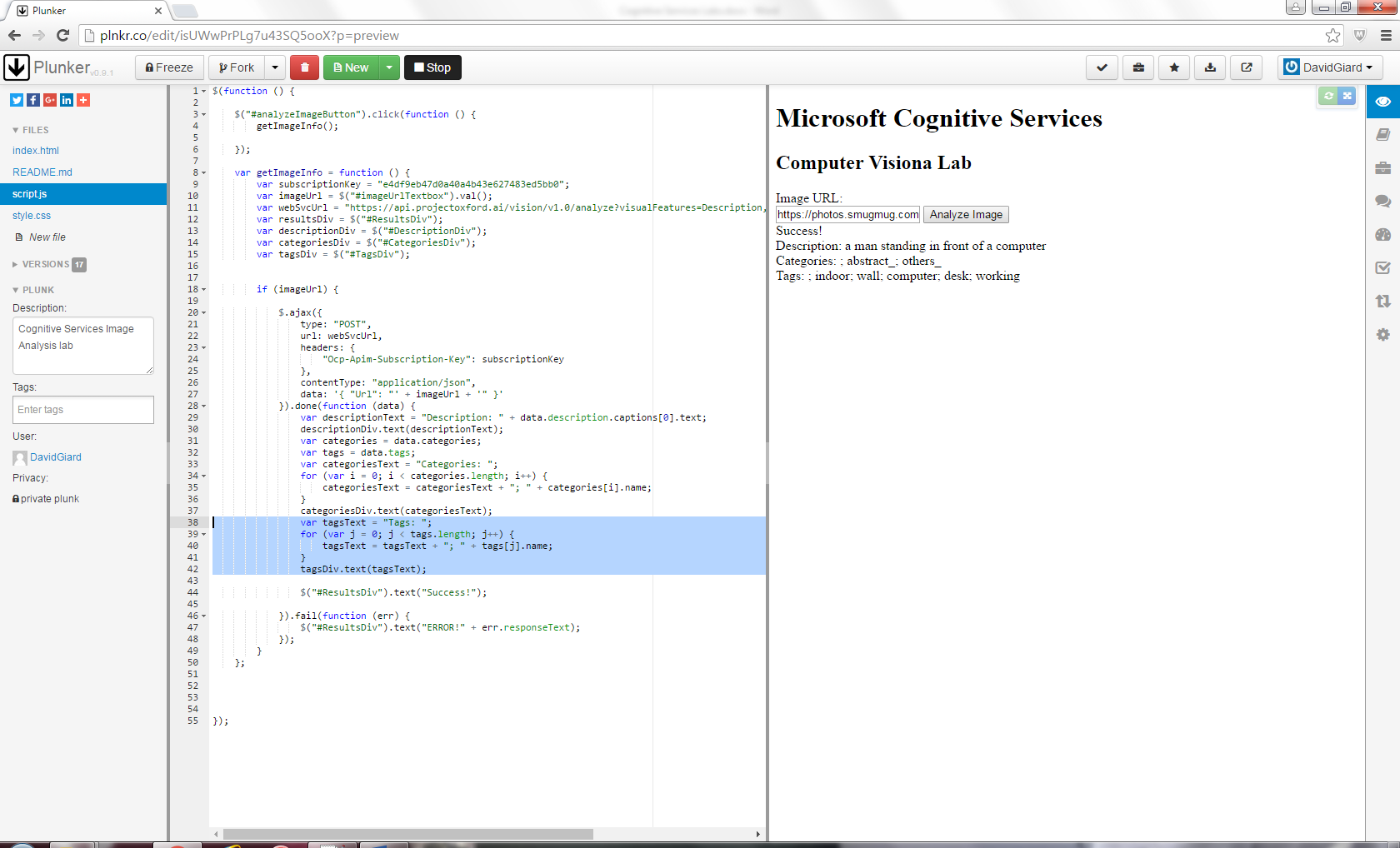
In the right toolbar, click the “Live Preview” button (). The rendering of your HTML will display in the right pane of the browser, as shown below:



Select “script.js” from the FILES on the right.

In the script.js, add the following JavaScript:

|  |
| --- |
| $(function () {  $("#analyzeImageButton").click(function () {  getImageInfo();  });  var getImageInfo = function () {  var subscriptionKey = "e4df9eb47d0a40a4b43e627483ed5bb0";  var imageUrl = $("#imageUrlTextbox").val();  var webSvcUrl = "https://api.projectoxford.ai/vision/v1.0/analyze?visualFeatures=Description,Categories,Tags ";  var resultsDiv = $("#ResultsDiv");  var descriptionDiv = $("#DescriptionDiv");  var categoriesDiv = $("#CategoriesDiv");  var tagsDiv = $("#TagsDiv");  if (imageUrl) {  $.ajax({  type: "POST",  url: webSvcUrl,  headers: {  "Ocp-Apim-Subscription-Key": subscriptionKey  },  contentType: "application/json",  data: '{ "Url": "' + imageUrl + '" }'  }).done(function (data) {  var descriptionText = "Description: " + data.description.captions[0].text;  descriptionDiv.text(descriptionText);  var categories = data.categories;  var tags = data.tags;  var categoriesText = "Categories: ";  for (var i = 0; i < categories.length; i++) {  categoriesText = categoriesText + "; " + categories[i].name;  }  categoriesDiv.text(categoriesText);  var tagsText = "Tags: ";  for (var j = 0; j < tags.length; j++) {  tagsText = tagsText + "; " + tags[j].name;  }  tagsDiv.text(tagsText);  $("#ResultsDiv").text("Success!");  }).fail(function (err) {  $("#ResultsDiv").text("ERROR!" + err.responseText);  });  }  };  }); |



In the Preview pane, click the “Analyze Image” button.

The Caption and a list of categories and tags should display beneath the text box.