

ASSIGNMENT COVERSHEET

Student Name: Arash Banihashemi	
Class: Website Design	
Assignment: A1 – "Walk&Learn" Website Development	
Lecturer: David Petryca	Semester: 1404
Due Date: 19/11/2014	Actual Submission Date: 19/11/2014

Evidence Produced (List separate items)	Location (Choose one)	
PDF	X	Uploaded to the Learning Center (Moodle)
		Submitted to reception
<i>Note: Email submissions to the lecturer are not valid.</i>		

Student Declaration	
I declare that the work contained in this assignment was researched and prepared by me, except where acknowledgement of sources is made. I understand that the college can and will test any work submitted by me for plagiarism. Note: The attachment of this statement on any electronically submitted assignments will be deemed to have the same authority as a signed statement	
Date: 19/11/2014	Student Signature: Arash Banihashemi

A separate feedback sheet will be returned to you after your work has been graded.
Refer to your Student Manual for the Appeals Procedure if you have concerns about the grading decision.

Student Comment (Optional)
Was the task clear? If not, how could it be improved?
Was there sufficient time to complete the task? If not, how much time should be allowed?
Did you need additional assistance with the assignment?
Was the lecturer able to help you?
Were there sufficient resources available?
How could the assignment be improved?
<i>For further comments, please use the reverse of this page.</i>

GOOGLE MAP API

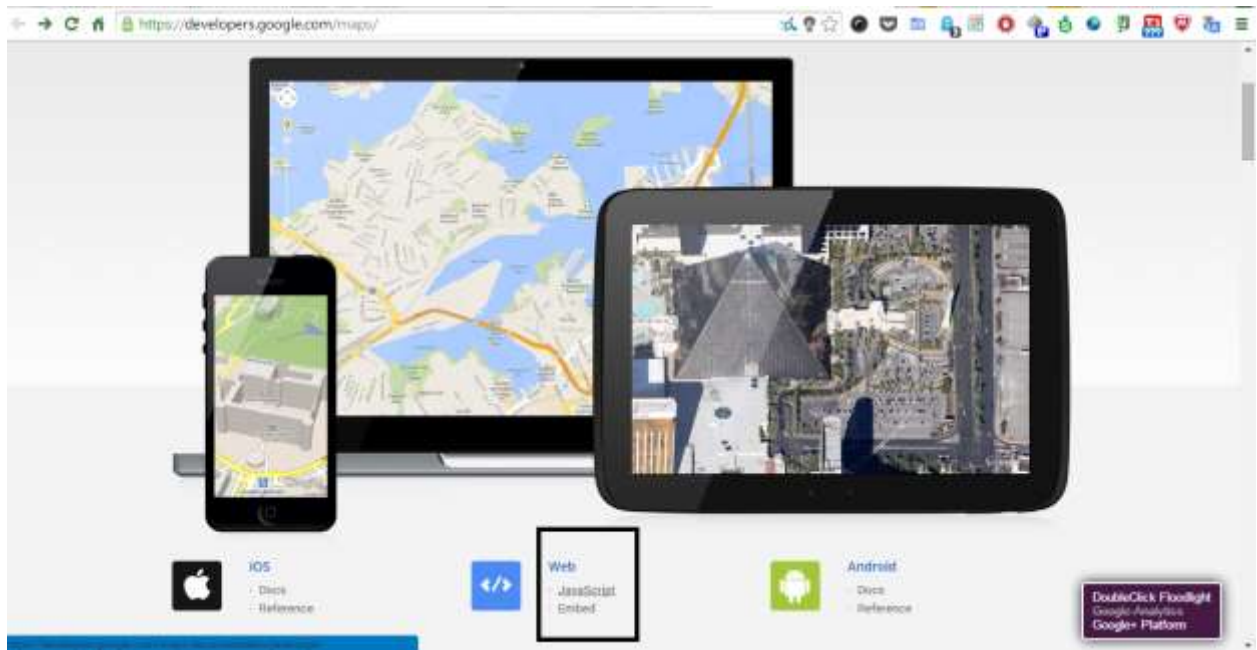
Arash Banihashemi
WEBSITE DESIGN

Contents

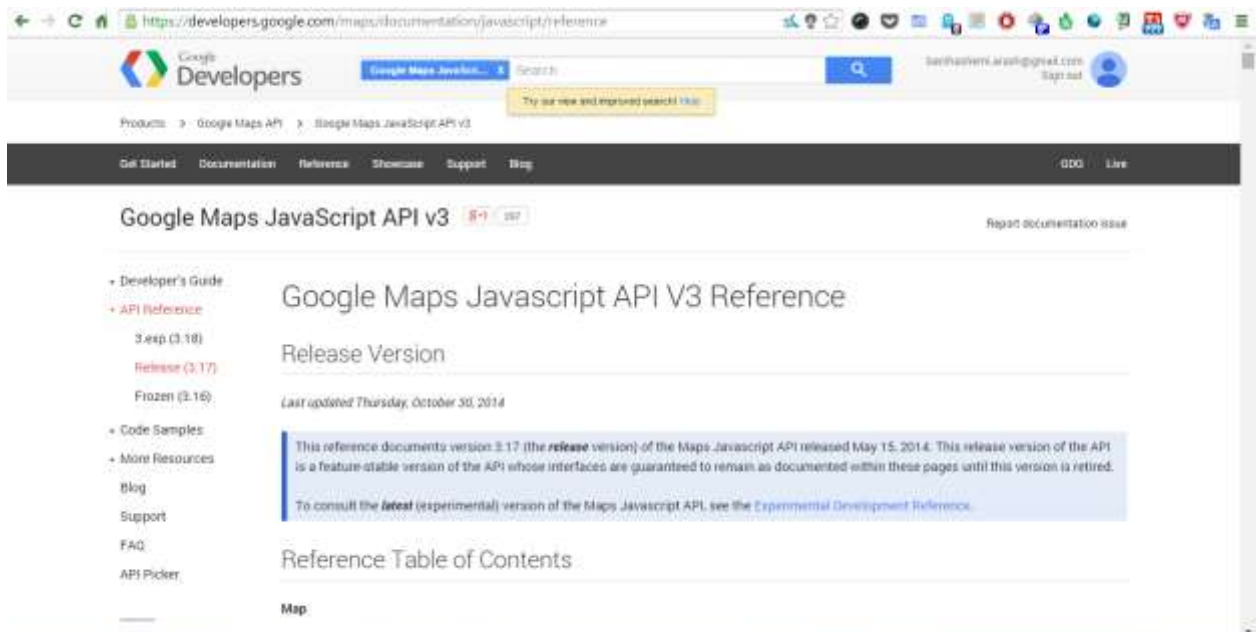
Introduction	2
Add a Marker:	6
Info Windows.....	7

Introduction

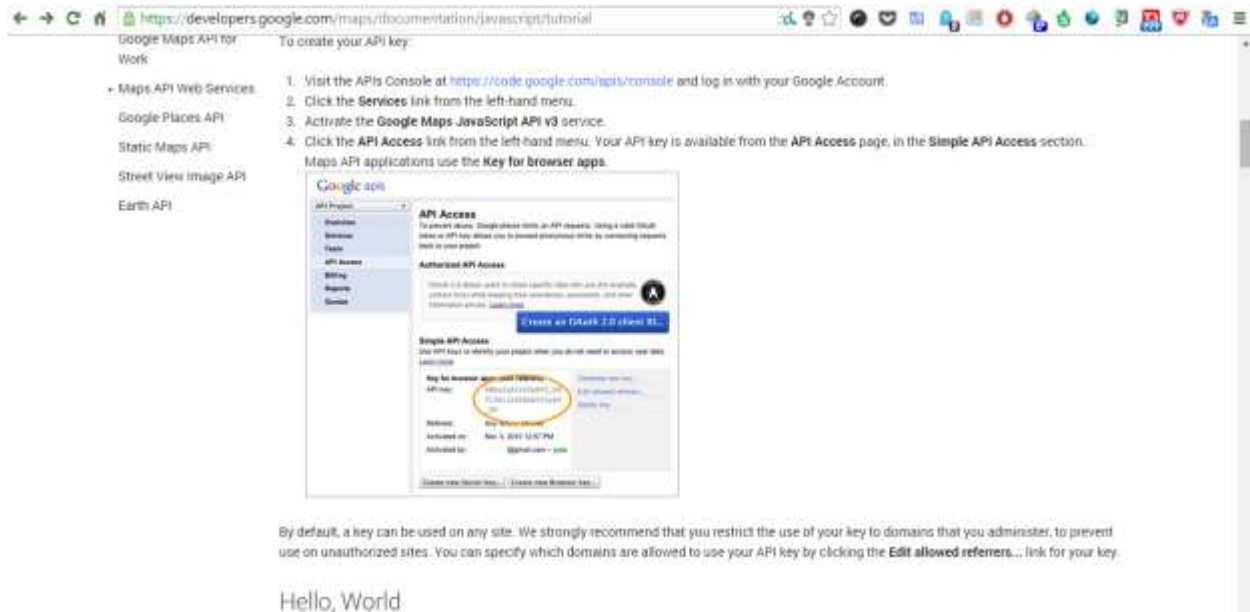
In this report I'm going to explain how to use google maps JavaScript API. To use Google map we need to setup an API key. The main site for google Maps documentation is <http://developers.google.com/maps>. In this URL we need to select our platform, for this report we use JavaScript for Web API then click on "Get Started" Link.



In "Get Started" page, there are list of documentation list. In the left Nav-Bar, Click on "API Reference".



This is where the actual API specification live and show how functions work. To use google maps we need to have API Key. The reason why we need API key is for us and google too keep track of how much we use this API.

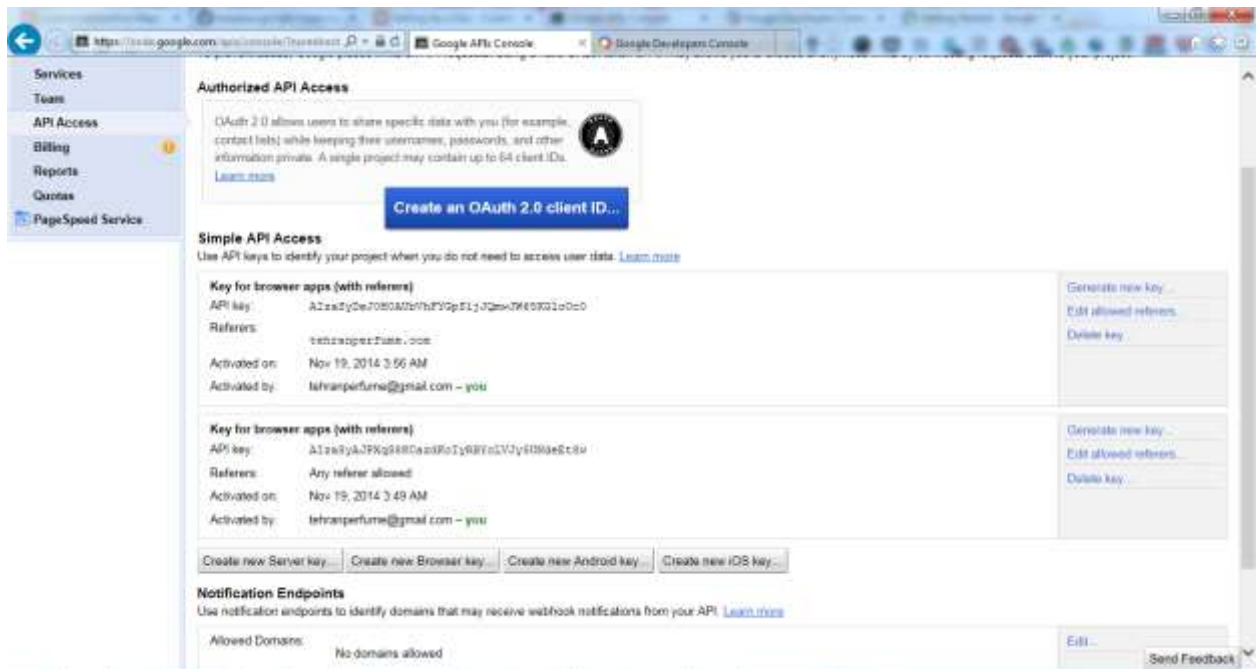


To create google API, we need to click on API's Console link then click on "Services" in the left nav-bar.

	Google Maps Coordinate API		<input type="checkbox"/>	Courtesy limit: 1,000 requests/day
	Google Maps Embed API		<input type="checkbox"/>	Courtesy limit: 2,000,000 requests/day
	Google Maps Engine API		<input type="checkbox"/>	Courtesy limit: 10,000 requests/day
	Google Maps Geolocation API		<input type="checkbox"/>	Courtesy limit: 0 requests/day • Pricing
	Google Maps JavaScript API v3		<input checked="" type="checkbox"/>	Courtesy limit: 25,000 requests/day • Pricing
	Google Maps SDK for iOS		<input type="checkbox"/>	
	Google Maps Tracks API		<input type="checkbox"/>	
	Google Mirror API		<input type="checkbox"/>	Courtesy limit: 1,000 requests/day
	Google Picker API		<input type="checkbox"/>	Courtesy limit: 10,000 requests/day

In this page "Google Maps JavaScript API v3" is currently OFF, click the off button to turn it ON.

To use google API key, Click on "API Access" then click on "Create New Browser Key...." Then write your domain address in the form.



Now, we have generated API key.

To use google map API we need to include google map script into index.html and inside body tag and also script.js like below codes:

Index.html :

```

1  <!DOCTYPE html>
2  <html>
3
4  <head>
5    <link rel="stylesheet" href="style.css">
6  </head>
7
8  <body>
9    <div id="map-canvas"></div>
10   <script src="https://maps.googleapis.com/maps/api/js?sensor=false">
11   </script>
12   <script src="script.js"></script>
13 </body>
14
15 </html>

```

The first thing we are going to do is create anonymous function in script.js. With anonymous function we can control the scope in your project and also import another library into our anonymous function. In this case we are going to import (window) as a parameter and also path through google name space.

To build a map we need map options and actual map itself. Map option is just an object literal.

Inside map options we need to put options like center and zoom. Center option is just simply center position of our map and we have two property inside center option: Latitude and longitude,

Inside zoom option, we need to put actual zoom level of the map.

Below the Map, this is where the google name space comes in.

```
map = new google.maps.Map();
```

This is the constructor and we need to path to parameter in this constructor like below code

```
map = new google.maps.Map(element, options);
```

We've already declared "options" and we need to declare element.

```
element = document.getElementById('map-canvas'),
```

In this step we need to create map-canvas inside html

```
<div id="map-canvas"></div>
```

In this step we need to set size of our map. To set size we need to put the inside css file like below code.

```
html {  
    height: 100%;  
}  
body {  
    height: 100%;  
    padding: 0;  
    margin: 0;  
}  
#map-canvas {  
    height: 100%;  
}
```

Add a Marker:

The Marker constructor creates a marker. (Note that the position property must be set for the marker to display).

Add the marker to the map by using the setMap() method:



```
var marker=new google.maps.Marker({  
  position:myCenter,  
});
```


Info Windows

An InfoWindow displays content (usually text or images) in a popup window above the map, at a given location. The info window has a content area and a tapered stem. The tip of the stem is attached to a specified location on the map.

```
var infowindow = new google.maps.InfoWindow({  
  content: "Hello World!"  
});
```

```
infowindow.open(map, marker);
```

