Working with Google Web Services

The Google Maps Embed API provides four **Modes** that makes it easy to include an interactive map in your webpages. The following table lists the three services and their purpose:

|  |  |
| --- | --- |
| Mode | Purpose |
| Google Maps Directions | display the path between two or more specified points on the map, as well as the distance and travel time. |
| Google Maps Place | display a map pin at a particular place or address, such as a landmark, business, geographic feature, or town. |
| Google Maps Search | display results for a search across the visible map region. |
| Google Maps StreetView | display Street View images on your site or blog as interactive panoramas. |

In the following exercises, you will be adding Google Web Services to a webpage to help users search for places of interest.

# Exercise #1: Link to external CSS & JavaScript files

Cascading Style Sheets (CSS) contain the style & layout instructions that affect the appearance of a webpage. JavaScript files contain functions that enable webpages to react to user actions like mouse hover, drag, & clicks. Complete the tasks in this exercise in order to enable all the features of the webpages.

This table lists HTML elements used in this exercise.

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Start tag | End tag | Purpose |
| Head | <head> | </head> | contains links to CSS & JavaScript files, etc. |
| Body | <body> | </body> | contains the visible elements of your webpage |
| Link | <link> |  | links to an external CSS file |
| Script | <script> | </script> | links to an external JavaScript file |

The syntax for these elements are:

**<body** *attributes* **>** *content* **</body>**

**<head > </head>**

**<script** *attributes* **> </script>**

**<link** *attributes* **>**

## Task #1: Link to an external CSS

Linking to an external CSS file allows you to use the styling & layout rules defined in the file. Use the ***link*** element to link your webpage to an external CSS file. The ***link*** element syntax is:

**<link** *attributes* **>**

Follow these steps to *link* your webpage to an external CSS file:

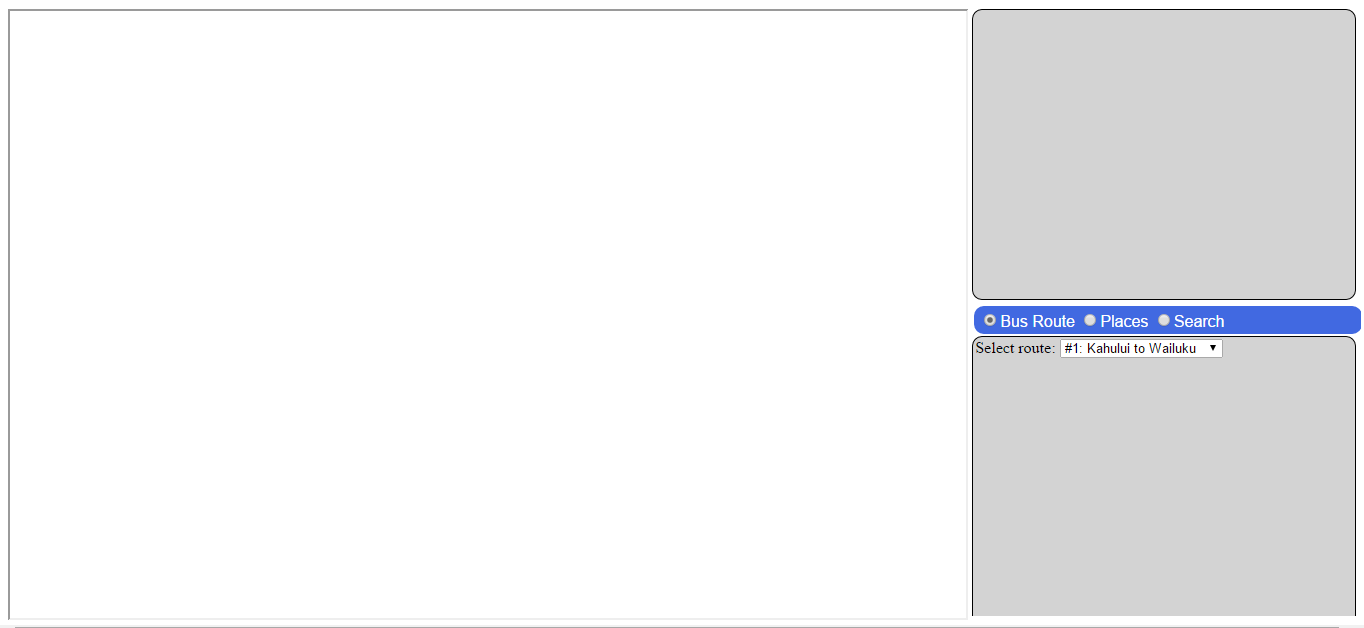
1. Open the file ***basic.html*** in your ***day2*** folder.
2. Preview the webpage before making any changes. Your webpage before should look like this:



1. Find the **head** element and add a ***link*** element to its content. Add the following attributes to the ***link*** element:

rel="stylesheet" type="text/css" href="css/gws.css"

1. Save your changes and refresh the webpage.
2. Verify that the ***style & layout*** of your webpage is similar to the figure below.



1. When you’ve completed this task, continue on with the next task.

## Task #2: Modify CSS styling rules.

You can refer t[o http://www.w3schools.com/cssref/default.asp](http://www.w3schools.com/cssref/default.asp) for more help with CSS rules.

Follow these steps to make changes to your CSS styling rules:

1. Open the file ***gws.css*** in your ***web/css*** folder.
2. Find the CSS rule for the **body** element.
3. Refer t[o http://www.w3schools.com/cssref/css\_colornames.asp](http://www.w3schools.com/cssref/css_colornames.asp) for a list of CSS color names.
4. Add the following changes to the #radio-menu > label:hover rule

box-shadow: 5px 5px 5px black;

color: black;

background-color: gold;

1. Save your changes.
2. Move your mouse over the button menu and view the results.
3. Experiment with other transform functions in the #radio-menu > label:hover rule

To rotate a button label 360 degrees, add the following transform:

transform: rotate(360deg);

To move a button button 20 pixels to the left, add the following transform:

transform: translate(20px);

To increase the size of a button label 2x, add the following transform:

transform: scale(2,2);

1. Refer t[o](http://www.w3schools.com/cssref/css_colornames.asp) <http://www.w3schools.com/cssref/css3_pr_transform.asp> for a list of available transforms.
2. When you’ve completed this task, continue on with the next task.

## Task #3: Include an external *JavaScript* file

The ***script*** element is used to include JavaScript functionality into a webpage. The ***script*** element syntax is:

**<script** *attributes* **> </script>**

Refer t[o http://www.w3schools.com/tags/tag\_script.asp](http://www.w3schools.com/tags/tag_script.asp) for help with using the ***script*** element.

Follow these steps to add JavaScript functions to your webpage:

1. Open the file ***basic.html*** in your ***day2*** folder.
2. Find the **head** element and add a ***script*** element to its content.

Add the following ***attributes*** to the **script** element:

type="text/javascript" src="js/gws.js"

1. Save your changes and refresh the webpage
2. Verify that your webpage looks similar to the figure below.

## 

## Task #4: Add Google Maps Directions to your webpage

The Google Maps ***Directions*** mode displays a map marker at a particular place or address, such as a landmark, business, geographic feature, or town. Refer to the ***Directions*** mode documentation for more help: <https://developers.google.com/maps/documentation/embed/guide>

When the user selects a route from the list, the ***showDirections*** function will:

* Construct a URL with the following syntax:

https://www.google.com/maps/embed/v1/directions?key=**API\_KEY**

&origin=Oslo+Norway

&destination=Telemark+Norway

The following URL parameters are required:

**origin**:  defines the starting point from which to display directions. The value can be either a place name or address. The string should be URL-escaped, so an address such as "City Hall, New York, NY" should be converted to City+Hall,New+York,NY. (The Google Maps Embed API supports both + and %20 when escaping spaces.)

**destination**:  defines the end point of the directions.

* Update the ***src*** attribute in the ***<iframe id=”gmMain”>*** element with the new URL

## Task #5: Add Google Maps Places to your webpage

The Google Embed Maps ***Place*** mode displays a map pin at a particular place or address, such as a landmark, business, geographic feature, or town. Refer to the ***Place*** mode documentation for more help: <https://developers.google.com/maps/documentation/embed/guide>

When the user clicks the **Find** button, the ***showPlace*** function will:

* Construct a URL with the following syntax:

https://www.google.com/maps/embed/v1/place?key=API\_KEY

&q=Fisht+Olympic+Stadium,Sochi+Russia

The following URL parameter is required:

**q**: defines the place to highlight on the map. It accepts a location as either a place name or address. The string should be URL-escaped, so an address such as "City Hall, New York, NY" should be converted to City+Hall,New+York,NY.

* Update the ***src*** attribute in the ***<iframe id=”gmMain”>*** element with the new URL

## Task #6: Add Google Maps Search to your webpage

The Google Embed Maps Search mode displays results for a search across the visible map region. It's recommended that a location for the search be defined, either by including a location in the search term (record+stores+in+Seattle) or by including a center and zoom parameter to bound the search. Refer to the Search mode documentation for more help: <https://developers.google.com/maps/documentation/embed/guide>

When the user clicks the **Search** button, the ***showSearch*** function will:

* Construct a URL with the following syntax:

https://www.google.com/maps/embed/v1/search?key=*API\_KEY*

&q=record+stores+in+Seattle

The following URL parameter is required:

**q:**  specifies the search term. It can include a geographic restriction, such as in+Seattle or near+98033.

* Update the ***src*** attribute in the ***<iframe id=”gmMain”>*** element with the new URL