Lifestyle Business Website

Statement of Work

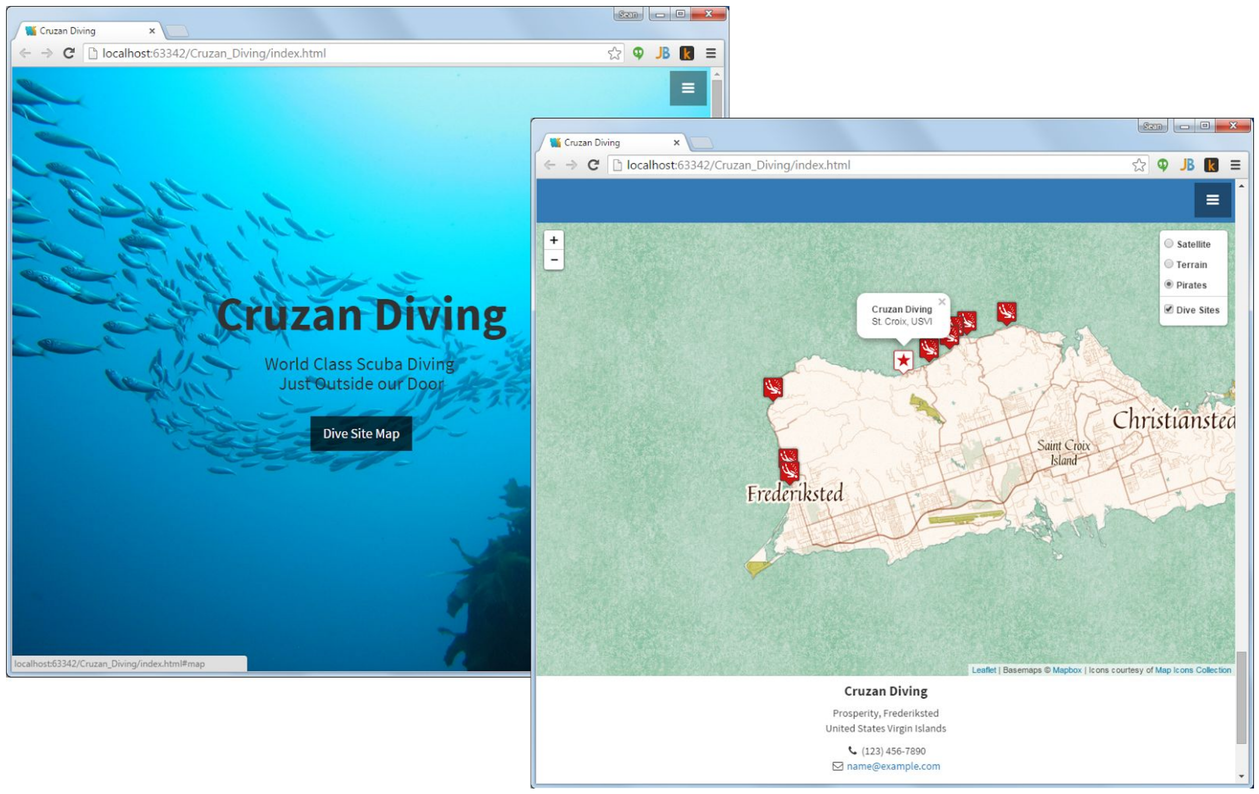
Project Team

Project Role	Organization	Contact	Email
GIS Developer	ACC	You	you@austincc.edu
Project Manager	ACC	Sean Moran	smoran@austincc.edu

Project Description

You won the lottery! That's the good news. The bad news is that it's not enough to quit working. Undaunted, you've decided to take the winnings and invest in a lifestyle business where you can make a living doing what you love. The tourism business you've purchased needs a facelift and a new website with an interactive map showing your location along with nearby attractions.

You're on a budget, so you've selected a low cost open source GIS technology stack that includes Leaflet, GeoJSON, and Mapbox to create your website and interactive map.



Responsive design website with interactive map of business location and nearby attractions

Duration
The project is expected to last two to four hours.

Type and Value
The Lifestyle Business Website is a school project worth 10 points toward the GISC 2459 final grade.

Payment
Payment will be extended in the form of a project grade.

Project Area
The project area will depend on the location of your business and nearby attractions.

Project Goal
Create and publish a lifestyle business website and interactive map using GeoJSON, Mapbox and the Leaflet JavaScript interactive mapping API.

Measures
The lifestyle business website should include:

1. A themed website template with homepage and link to an interactive map; and
2. An embedded interactive map with location of your business and nearby attractions.

The embedded interactive map should:

1. Utilize the Leaflet API;
2. Map your business location as a Leaflet marker and nearby attractions as GeoJSON features with attribute popups;
3. Include Mapbox cached basemap services; and
4. Include a toggle layer control.

The website and map should be published using an Amazon Web Service (AWS) Elastic Cloud Computing (EC2) virtual server with Microsoft IIS and be accessible via an http address to be submitted via Blackboard.

Project Scope
The project scope consists of the following summary tasks:

1. Create Website
2. Collect and Assimilate Data
3. Add Leaflet Map to Website
4. Add Layers to Leaflet Map
5. Publish Website with Embedded Map

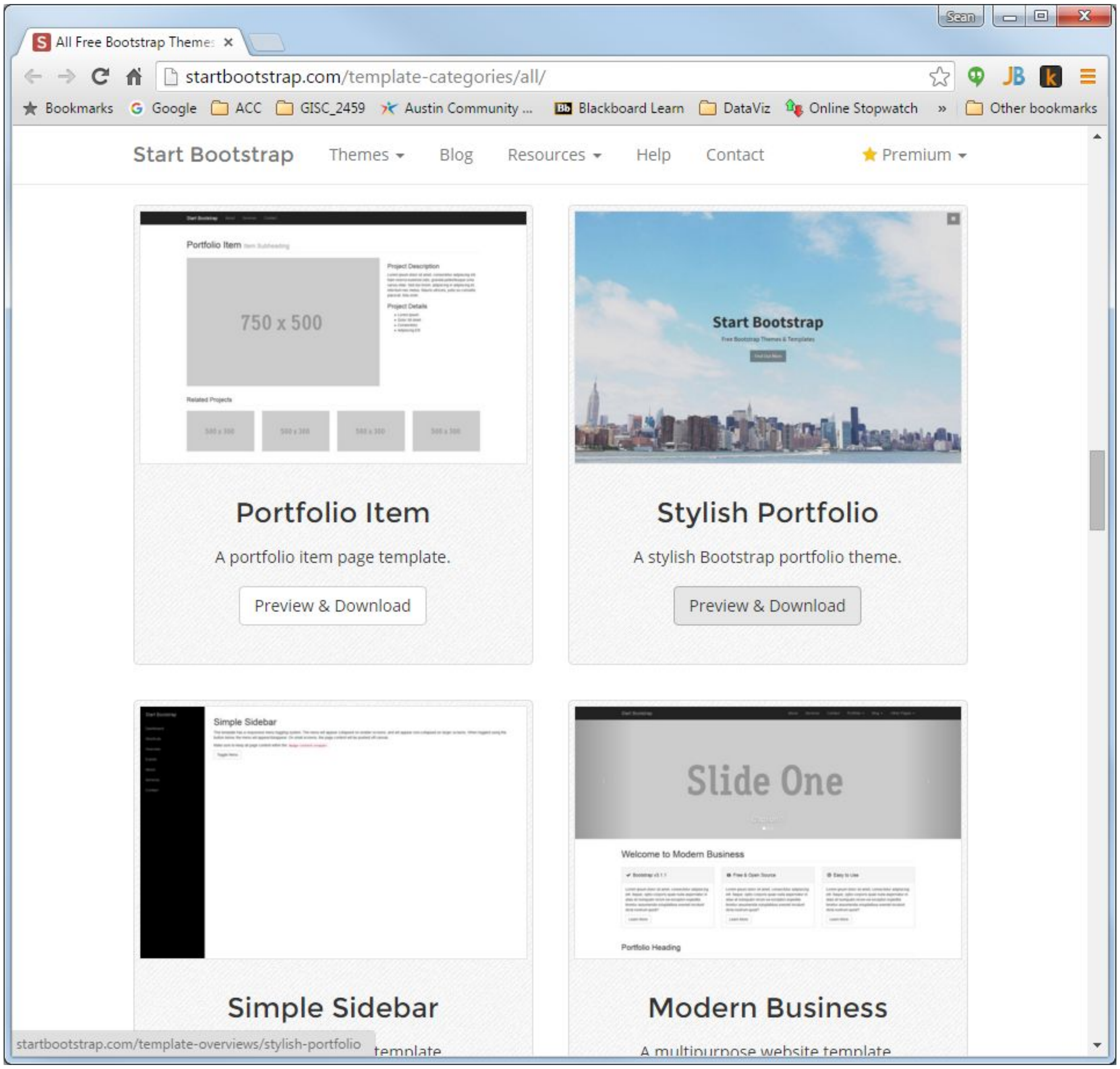
Each summary task is described below.

1. Create Website

A lot of your customers rely heavily on their mobile phones, so you want to use a responsive design website template. Create the website by selecting, downloading, and customizing an open source responsive design website template from Iron Summit Media Strategies' Start Bootstrap website at http://startbootstrap.com/.

Select, Download, and Copy Website Template

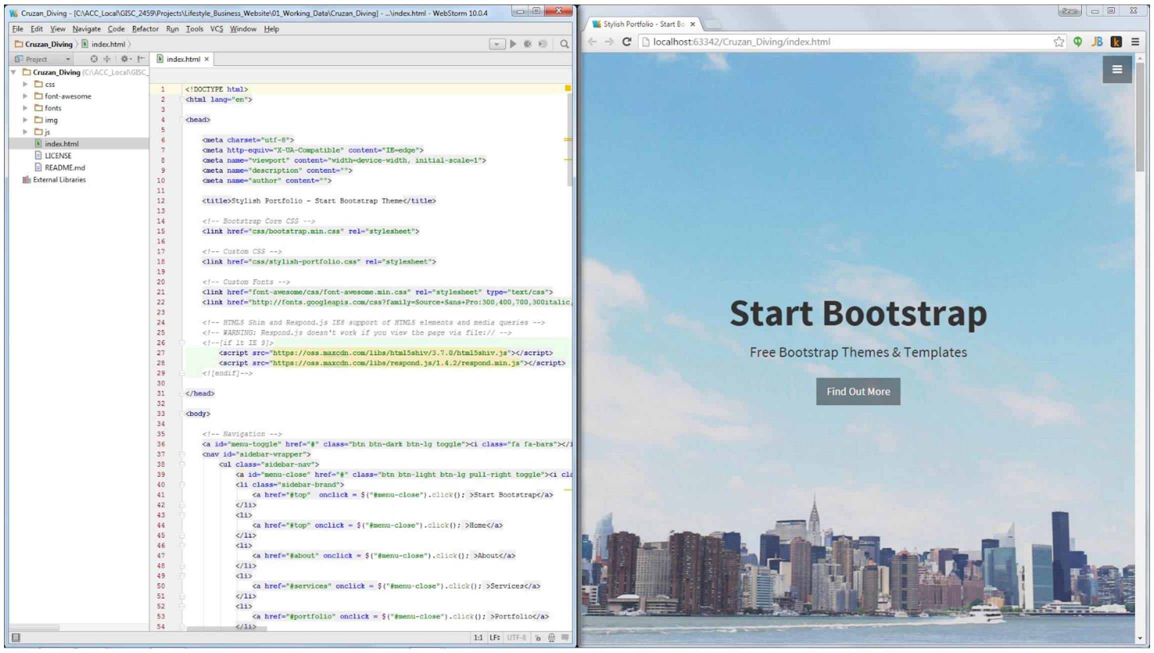
Select and download a website template from http://startbootstrap.com/template-categories/all/. Make a working copy that allows you to experiment with the template without having to return to Start Bootstrap website if you "break" it.



Start Bootstrap responsive design website templates from Iron Summit Media Strategies

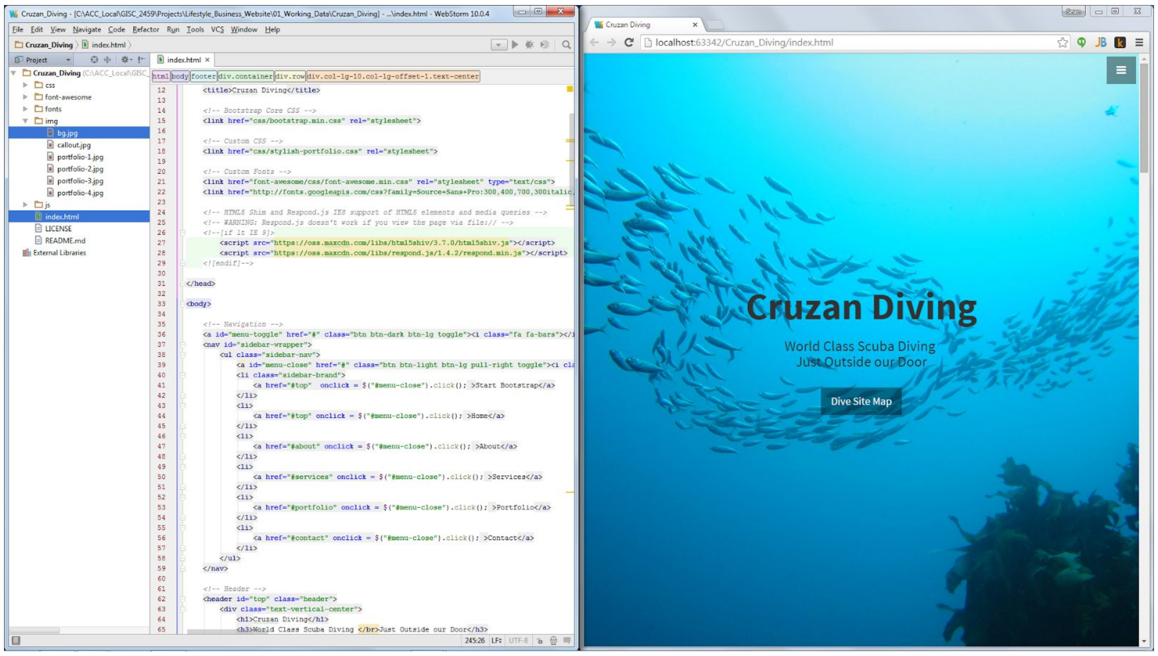
Open and Edit Website Template in an IDE

Utilize an Integrated Development Environment (IDE), such as JetBrains WebStorm to open and edit the website. Customize the website home page title, heading, and related elements so that your customers identify the website with your lifestyle business. Remember to reserve a location for your interactive map.


Cruzan_Diving - [C:\ACC_Local\GISC_2459\Projects\Lifestyle_Business_Website\01_Working_Data\Cruzan_Diving] - ...\index.html - WebStorm 10.0.4
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Cruzan_Diving
index.html
Project
Cruzan_Diving (C:\ACC_Local\GISC_
css
font-awesome
fonts
img
js
index.html
LICENSE
README.md
External Libraries
1:1 LF: UTF-8
Stylish Portfolio - Start Bo
localhost:63342/Cruzan_Diving/index.html
Start Bootstrap
Free Bootstrap Themes & Templates
Find Out More

```html
<!DOCTYPE html>
<html lang="en">

<head>

    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">

    <title>Stylish Portfolio - Start Bootstrap Theme</title>

    <!-- Bootstrap Core CSS -->
    <link href="css/bootstrap.min.css" rel="stylesheet">

    <!-- Custom CSS -->
    <link href="css/stylish-portfolio.css" rel="stylesheet">

    <!-- Custom Fonts -->
    <link href="font-awesome/css/font-awesome.min.css" rel="stylesheet" type="text/css">
    <link href="http://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,700,300italic,

    <!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries -->
    <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
    <!--[if lt IE 9]>
        <script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
        <script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
    <![endif]-->

</head>

<body>

    <!-- Navigation -->
    <a id="menu-toggle" href="#" class="btn btn-dark btn-lg toggle"><i class="fa fa-bars"></i
    <nav id="sidebar-wrapper">
        <ul class="sidebar-nav">
            <a id="menu-close" href="#" class="btn btn-light btn-lg pull-right toggle"><i cla
            <li class="sidebar-brand">
                <a href="#top"  onclick = $("#menu-close").click(); >Start Bootstrap</a>
            </li>
            <li>
                <a href="#top" onclick = $("#menu-close").click(); >Home</a>
            </li>
            <li>
                <a href="#about" onclick = $("#menu-close").click(); >About</a>
            </li>
            <li>
                <a href="#services" onclick = $("#menu-close").click(); >Services</a>
            </li>
            <li>
                <a href="#portfolio" onclick = $("#menu-close").click(); >Portfolio</a>
            </li>
```

The Start Bootstrop Stylish Portfolio website template opened in WebStorm IDE and Internet browser


Cruzan_Diving - [C:\ACC_Local\GISC_2459\Projects\Lifestyle_Business_Website\01_Working_Data\Cruzan_Diving] - ...\index.html - WebStorm 10.0.4
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Cruzan_Diving
index.html
Project
Cruzan_Diving (C:\ACC_Local\GISC_
css
font-awesome
fonts
img
bg.jpg
callout.jpg
portfolio-1.jpg
portfolio-2.jpg
portfolio-3.jpg
portfolio-4.jpg
js
index.html
LICENSE
README.md
External Libraries
html body footer div.container div.row div.col-lg-10.col-lg-offset-1.text-center
245:26 LF: UTF-8
Cruzan Diving
localhost:63342/Cruzan_Diving/index.html
Cruzan Diving
World Class Scuba Diving
Just Outside our Door
Dive Site Map

```html
    <title>Cruzan Diving</title>

    <!-- Bootstrap Core CSS -->
    <link href="css/bootstrap.min.css" rel="stylesheet">

    <!-- Custom CSS -->
    <link href="css/stylish-portfolio.css" rel="stylesheet">

    <!-- Custom Fonts -->
    <link href="font-awesome/css/font-awesome.min.css" rel="stylesheet" type="text/css">
    <link href="http://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,700,300italic,

    <!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries -->
    <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
    <!--[if lt IE 9]>
        <script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
        <script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
    <![endif]-->

</head>

<body>

    <!-- Navigation -->
    <a id="menu-toggle" href="#" class="btn btn-dark btn-lg toggle"><i class="fa fa-bars"></i
    <nav id="sidebar-wrapper">
        <ul class="sidebar-nav">
            <a id="menu-close" href="#" class="btn btn-light btn-lg pull-right toggle"><i cla
            <li class="sidebar-brand">
                <a href="#top"  onclick = $("#menu-close").click(); >Start Bootstrap</a>
            </li>
            <li>
                <a href="#top" onclick = $("#menu-close").click(); >Home</a>
            </li>
            <li>
                <a href="#about" onclick = $("#menu-close").click(); >About</a>
            </li>
            <li>
                <a href="#services" onclick = $("#menu-close").click(); >Services</a>
            </li>
            <li>
                <a href="#portfolio" onclick = $("#menu-close").click(); >Portfolio</a>
            </li>
            <li>
                <a href="#contact" onclick = $("#menu-close").click(); >Contact</a>
            </li>
        </ul>
    </nav>

    <!-- Header -->
    <header id="top" class="header">
        <div class="text-vertical-center">
            <h1>Cruzan Diving</h1>
            <h3>World Class Scuba Diving </br>Just Outside our Door</h3>
```

The website template with customized title, heading, and other elements

2. Collect and Assimilate Data

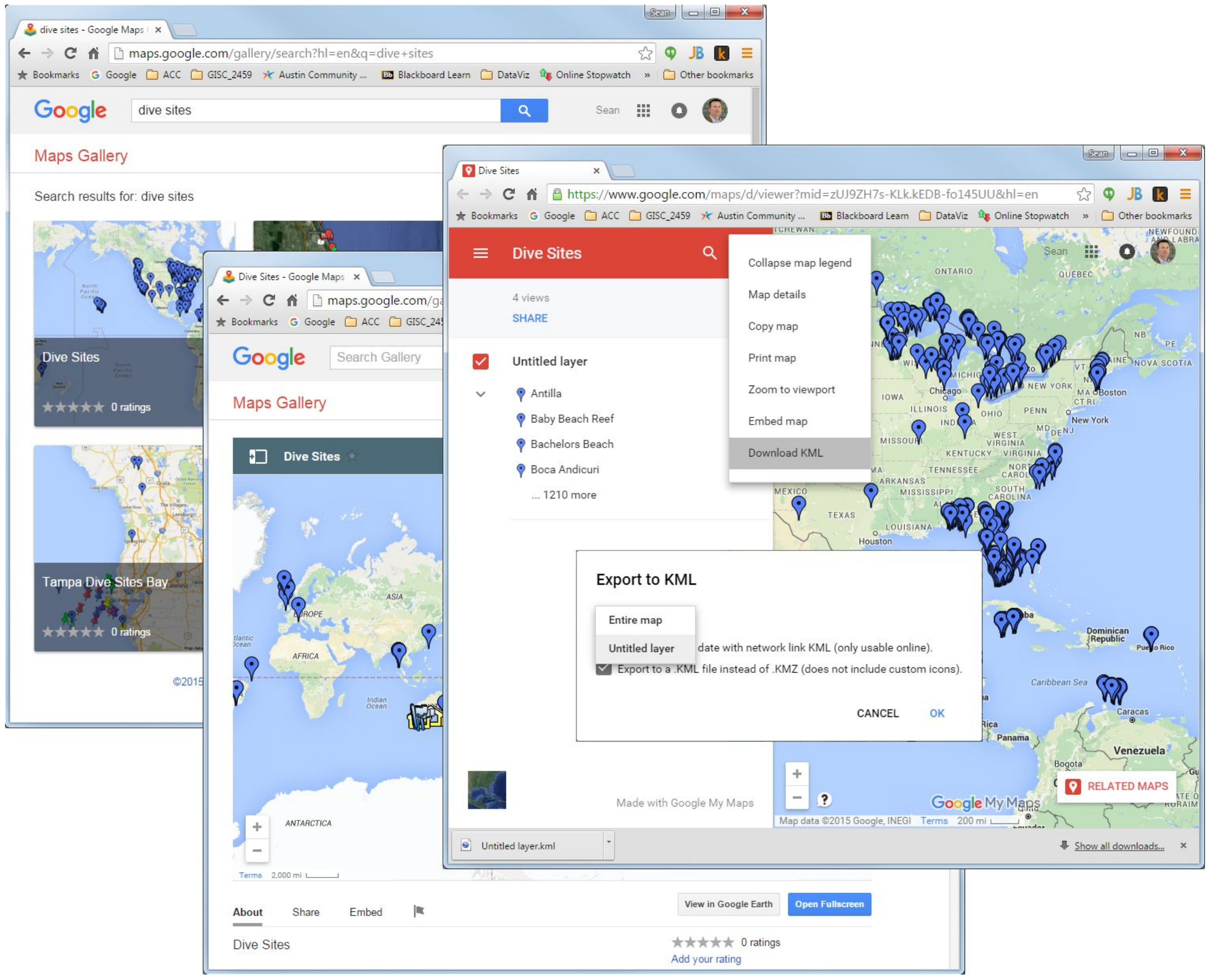
The interactive map will include your business location mapped as a Leaflet marker and nearby attractions mapped using GeoJSON with attribute popups. Start by finding a good location for your business and then locate nearby attractions for your customers to enjoy.

Locate Your Business

Use Google Maps to explore your market and locate your business. Once you identify a good location, right-click on the map and select "What's here?". A small window will popup with the latitude and longitude coordinates for your business. Copy and save these coordinates to use during Summary Task 3 Add Leaflet Map to Website.

Find and Download Geospatial Attractions Dataset

Now you need to locate nearby attractions for your customers. You can search google using the modifier "kml" to find a map or with attractions you are looking for (e.g. diving, hiking, surfing, etc.).



Geospatial attractions dataset downloaded from Google Maps Gallery as a KML file

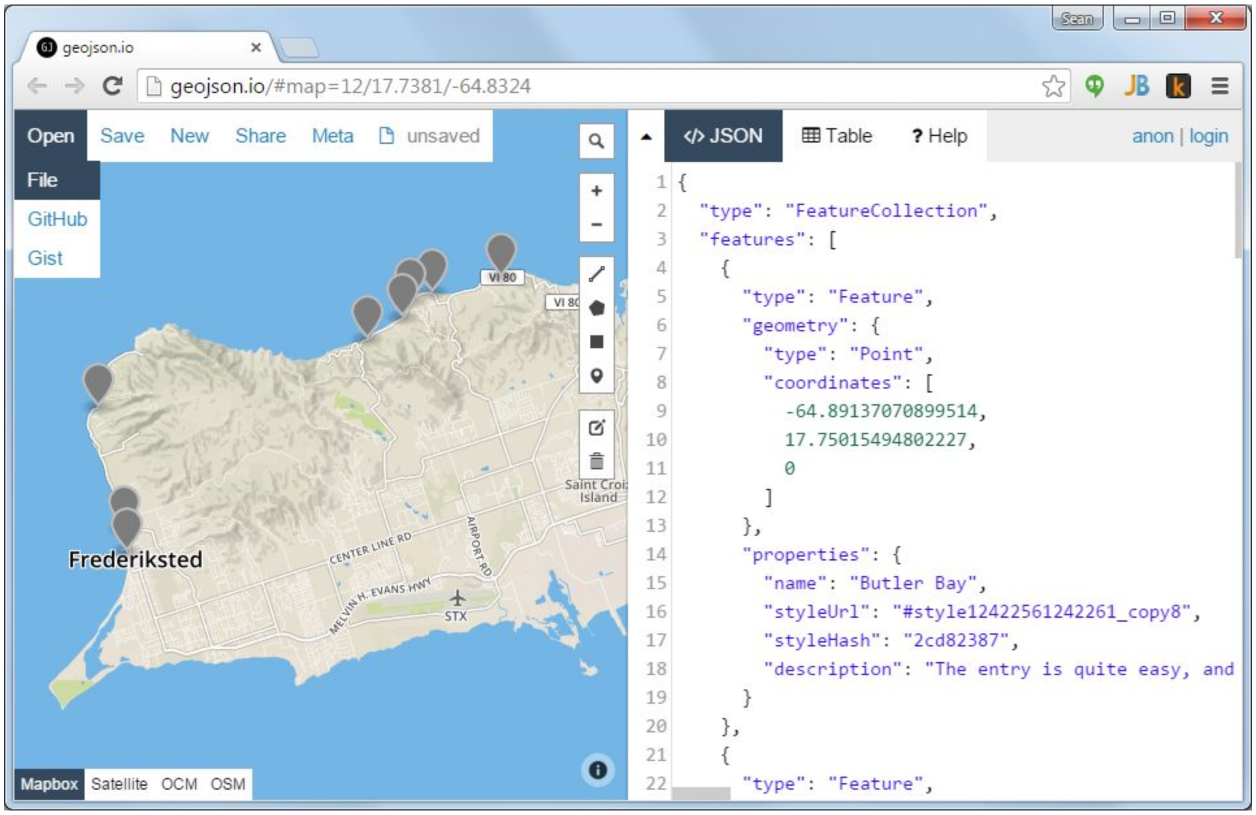
If necessary, you can use Google Earth to select a subset of features and and then Save Place As… to create a new KML. Optionally, you might locate and download your geospatial attractions in a different file format from a different site or create the data yourself.

Convert File to GeoJSON

Convert your geospatial attractions to a GeoJSON file. GeoJSON is an open standard JavaScript format for geographic features and attributes. Because it's JavaScript, you can serve geospatial data directly to your website and map without a map server. There are a number of online conversion tools you can use, including:

- Ogre - Click Choose File to select your KML, KMZ, CSV, GPX, Shapefile or other file format, click CONVERT TO GEOJSON, and copy GeoJSON from your browser window. With Ogre, you can also convert a GeoJSON file to Shapefile.
- geojson.io - Click Open>File to select your KML, CSV, GPX, or other file format and copy the GeoJSON from the right-side window or click Save>GeoJSON to download.
- toGeoJSON - open your KML or GPX file in notepad, copy the contents, paste them into the toGeoJSON left-side window, select the format, and then copy the GeoJSON from the right-side window or click the blue button to download.

You will use your GeoJSON of nearby attractions when you Add Layers to Leaflet Map in Summary Task 4.



```
{
  "type": "FeatureCollection",
  "features": [
    {
      "type": "Feature",
      "geometry": {
        "type": "Point",
        "coordinates": [
          -64.89137070899514,
          17.75015494802227,
          0
        ]
      },
      "properties": {
        "name": "Butler Bay",
        "styleUrl": "#style12422561242261_copy8",
        "styleHash": "2cd82387",
        "description": "The entry is quite easy, and
      }
    },
    {
      "type": "Feature",
```

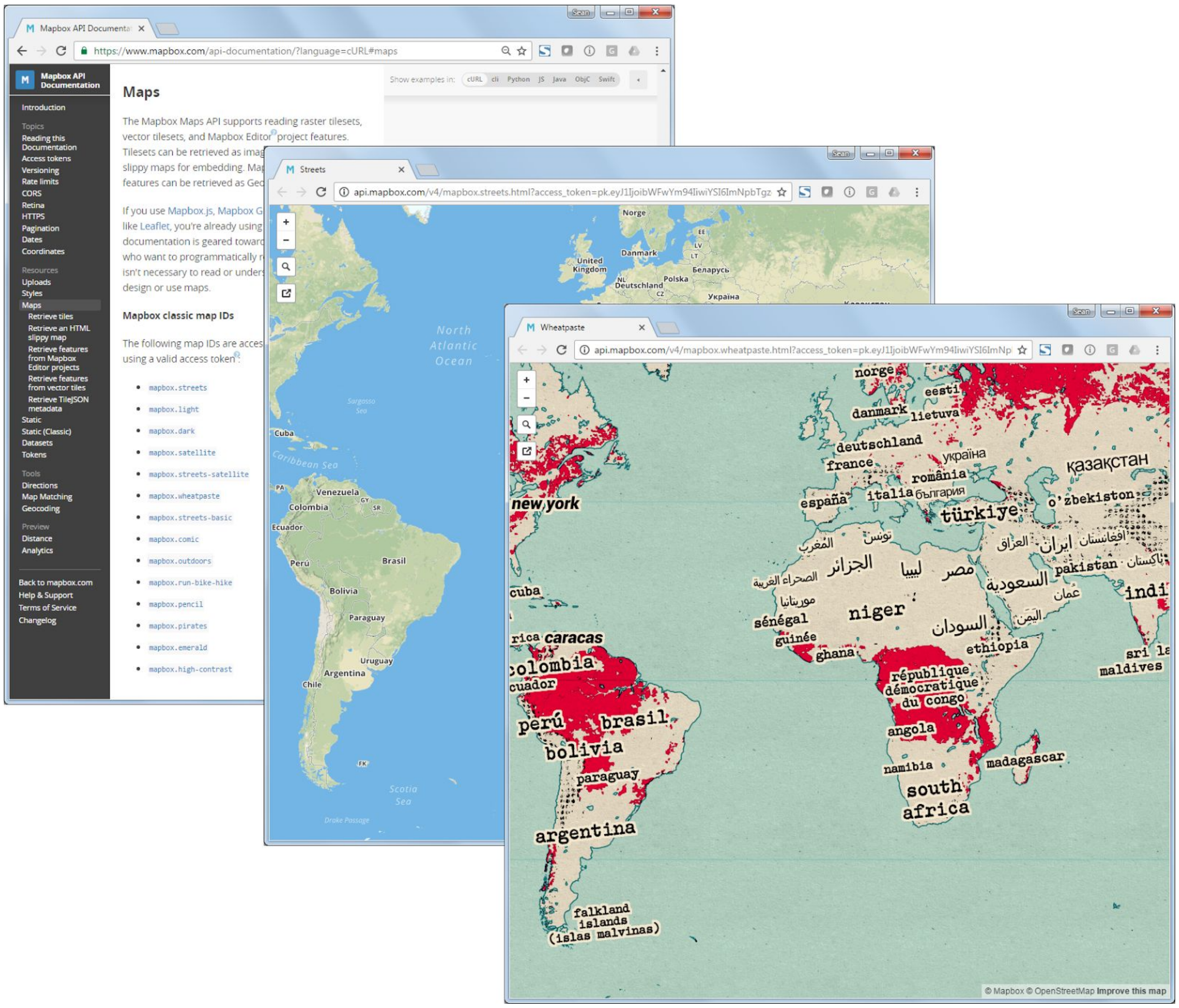
Geospatial attractions converted to GeoJSON using geojson.io website

3. Add Leaflet Map to Website

In Summary Task 1 Create Website, you created a website with a spot reserved for your embedded interactive map. The next step is to add the Leaflet API with Mapbox cached map services to your website. First, you need to create a Mapbox API Access Token and identify a map service that you'll call from the Leaflet API.

Create Mapbox API Access Token and Select Map Service

Mapbox has published a number of beautiful cached map services that developers with less than 50,000 map views per month can use free of charge. The Leaflet API consumes the Mapbox cached map services with an Access Token and a Map ID that identifies the cached map services shown below. Select a map service and copy the Map ID (e.g. mapbox.satellite).


Mapbox API Documentation
https://www.mapbox.com/api-documentation/?language=cURL#maps
Introduction
Topics
Reading this Documentation
Access tokens
Versioning
Rate limits
CORS
Retina
HTTPS
Pagination
Dates
Coordinates
Resources
Uploads
Styles
Maps
Retrieve tiles
Retrieve an HTML slippy map
Retrieve features from Mapbox Editor projects
Retrieve features from vector tiles
Retrieve TileJSON metadata
Static
Static (Classic)
Datasets
Tokens
Tools
Directions
Map Matching
Geocoding
Preview
Distance
Analytics
Back to mapbox.com
Help & Support
Terms of Service
Changelog
Streets
api.mapbox.com/v4/mapbox.streets.html?access_token=pk.eyJ1IjoibWFwYm94IiwiYSI6ImNpbTgz
Wheatpaste
api.mapbox.com/v4/mapbox.wheatpaste.html?access_token=pk.eyJ1IjoibWFwYm94IiwiYSI6ImNp
North Atlantic Ocean
© Mapbox © OpenStreetMap Improve this map

Maps

The Mapbox Maps API supports reading raster tilesets, vector tilesets, and Mapbox Editor project features. Tilesets can be retrieved as imag slippy maps for embedding. Ma features can be retrieved as Ge

If you use Mapbox.js, Mapbox G like Leaflet, you're already using documentation is geared towar who want to programmatically r isn't necessary to read or unders design or use maps.

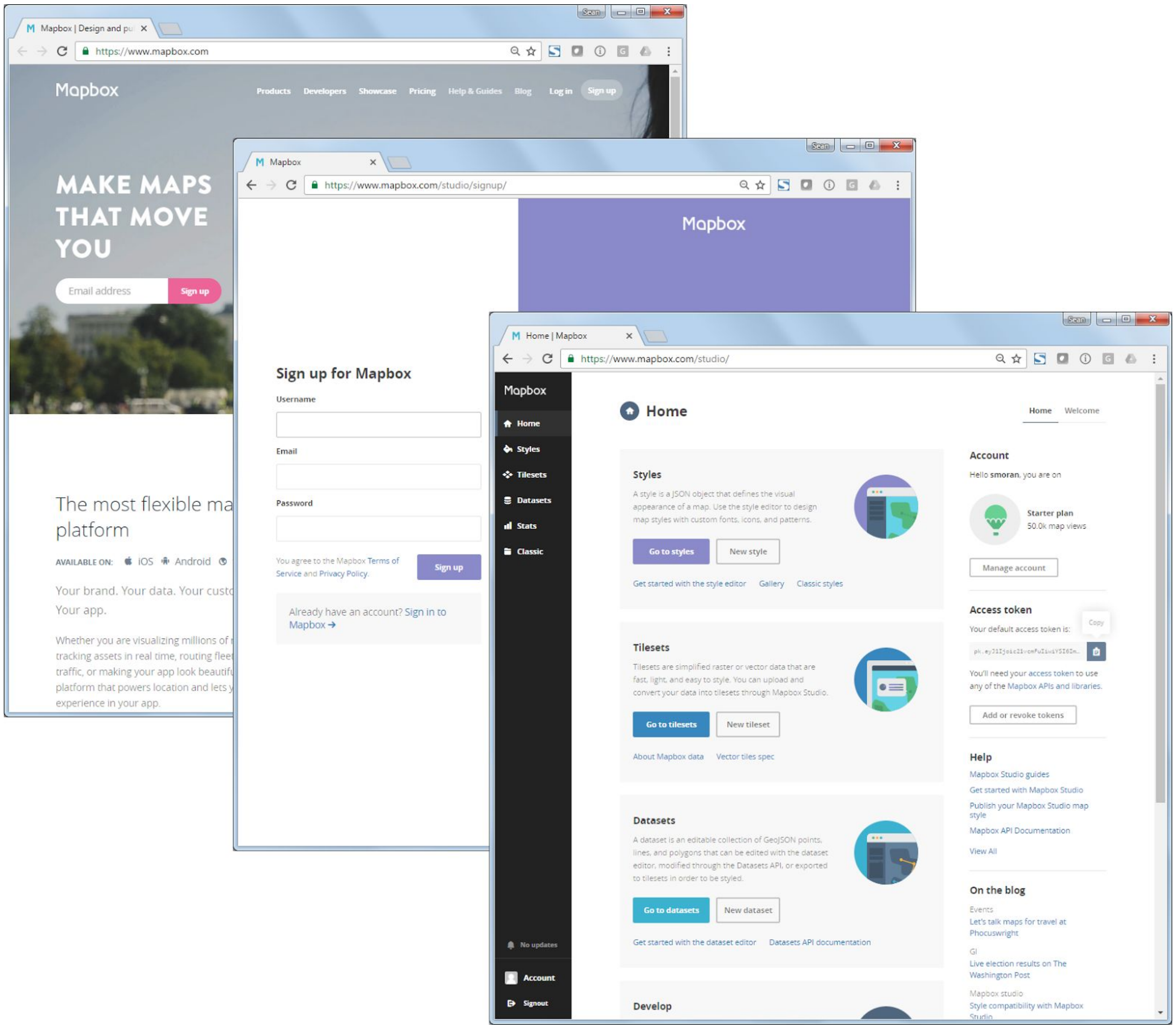
Mapbox classic map IDs

The following map IDs are acces using a valid access token:

- `mapbox.streets`
- `mapbox.light`
- `mapbox.dark`
- `mapbox.satellite`
- `mapbox.streets-satellite`
- `mapbox.wheatpaste`
- `mapbox.streets-basic`
- `mapbox.comic`
- `mapbox.outdoors`
- `mapbox.run-bike-hike`
- `mapbox.pencil`
- `mapbox.pirates`
- `mapbox.emerald`
- `mapbox.high-contrast`

Mapbox cached map services with Map IDs (e.g. mapbox.streets, mapbox.wheatpaste, etc.)

If you don't already have a Mapbox account, visit Mapbox at https://www.mapbox.com/ to Sign Up for an account, click on Projects, and copy your Access Token as shown below.



The most flexible ma
platform

AVAILABLE ON: iOS Android

Your brand. Your data. Your custo
Your app.

Whether you are visualizing millions of
tracking assets in real time, routing flee
traffic, or making your app look beautif
platform that powers location and lets y
experience in your app.

You agree to the Mapbox Terms of Service and Privacy Policy.

Already have an account? Sign in to Mapbox →

Styles

A style is a JSON object that defines the visual appearance of a map. Use the style editor to design map styles with custom fonts, icons, and patterns.

Tilesets

Tilesets are simplified raster or vector data that are fast, light, and easy to style. You can upload and convert your data into tilesets through Mapbox Studio.

Datasets

A dataset is an editable collection of GeoJSON points, lines, and polygons that can be edited with the dataset editor, modified through the Datasets API, or exported to tilesets in order to be styled.

Account

Hello **smoran**, you are on

Access token

Your default access token is:

pk.eyJ1Ijoic21vcmFuIiwiYSI6Im…

You'll need your access token to use any of the Mapbox APIs and libraries.

Help

On the blog

Events
Let's talk maps for travel at Phocuswright

GI
Live election results on The Washington Post

Mapbox studio
Style compatibility with Mapbox Studio

Using the Mapbox cached map services requires a Mapbox API access token

Now that you have a Mapbox API Access Token and a selected cached map service Map ID, you are ready to create your first Leaflet interactive map. The steps listed below are loosely based on the [Leaflet Quick Start Guide](#) found in the [Leaflet Tutorials](#).

Create Leaflet test map

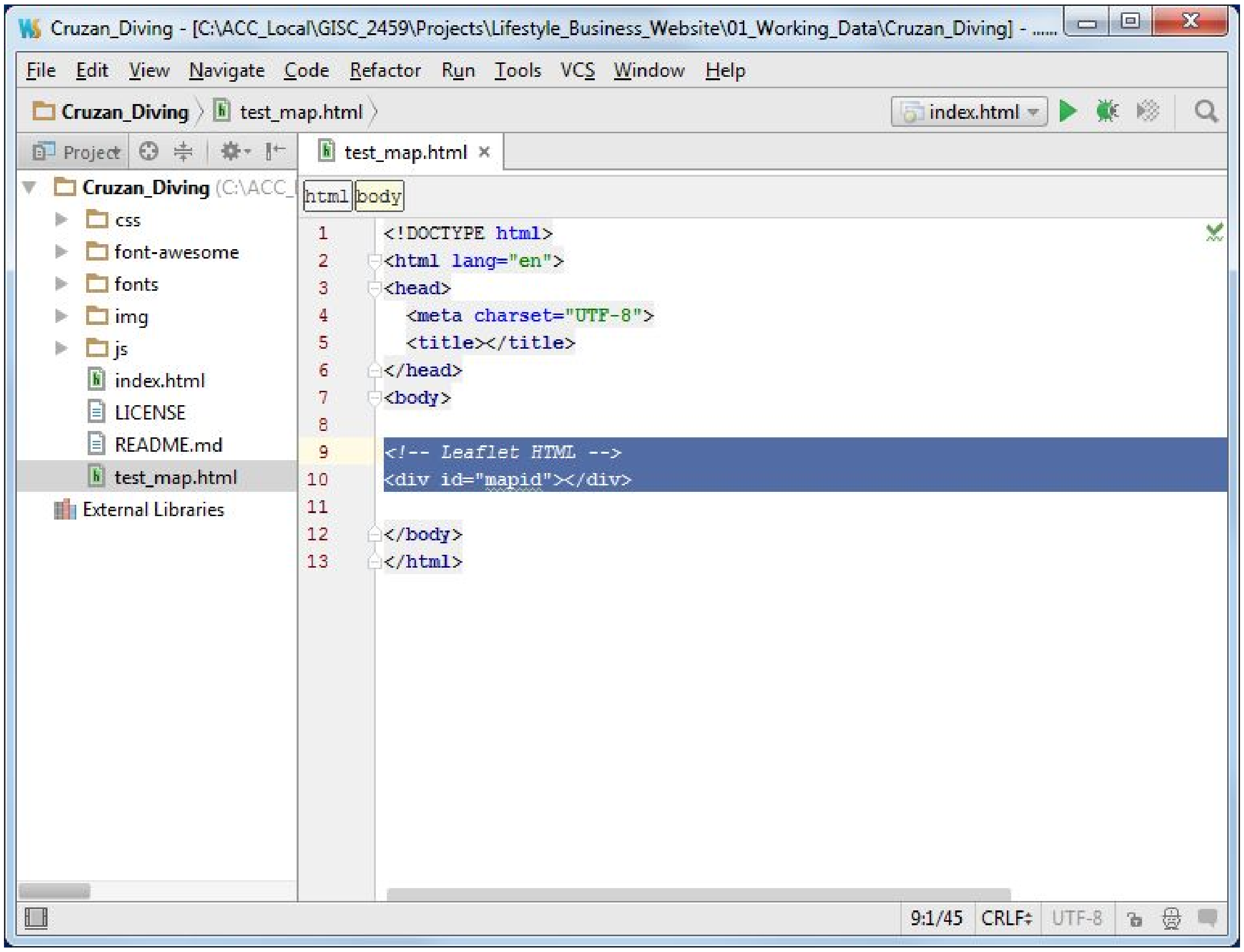
Good programmers test their code in a controlled environment before adding it to their website. You'll create a "test" HTML file to test your code before adding the working code blocks to your website.

The general approach for adding the Leaflet map to the website will be:

a. Add the HTML element
b. Add the variable declarations and JavaScript function
c. Add the CSS styling

Add the HTML Leaflet map element

Add the HTML Leaflet <div> element identified with the property id "mapid". The "mapid" property will be used by the JavaScript to place the map in this <div> container.


Cruzan_Diving - [C:\ACC_Local\GISC_2459\Projects\Lifestyle_Business_Website\01_Working_Data\Cruzan_Diving] - ......
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Cruzan_Diving test_map.html
index.html
Project
test_map.html
Cruzan_Diving (C:\ACC_
css
font-awesome
fonts
img
js
index.html
LICENSE
README.md
test_map.html
External Libraries
html body
9:1/45 CRLF UTF-8

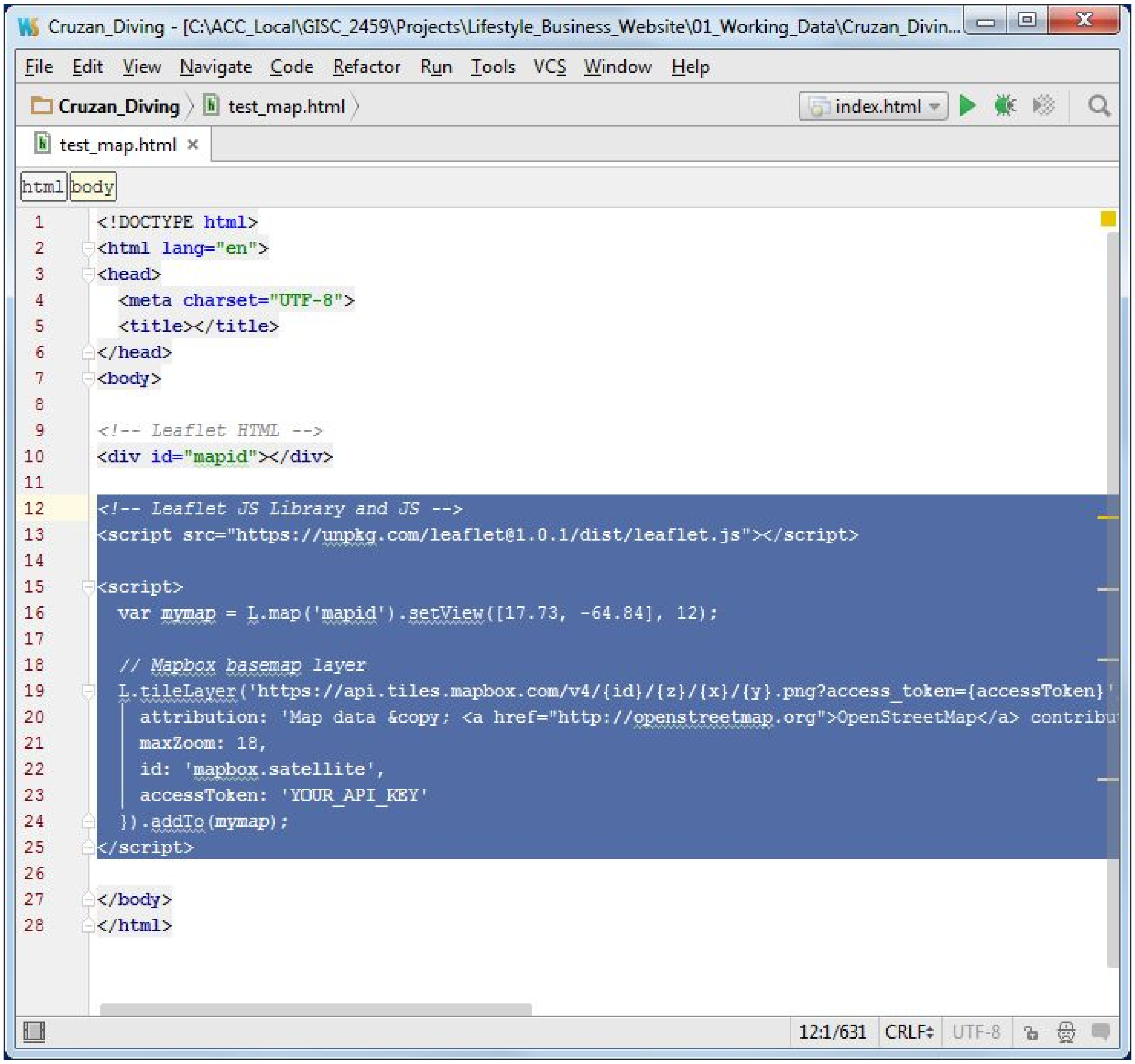
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title></title>
</head>
<body>

<!-- Leaflet HTML -->
<div id="mapid"></div>

</body>
</html>
```

test_map.html with HTML Leaflet <div> element

Add the Leaflet JavaScript file, variable declarations, and initialize map

Now you're ready to add the Leaflet JavaScript file, variable declarations and initialize the map. Loading the Leaflet JavaScript library allows you to use Leaflet API commands like L.tilelayer and L.map. You'll then declare a layer variable that includes the L.tilelayer command, Mapbox Access Token and Map ID. Lastly, you'll initialize the map using the L.map command with properties set to center, zoom, and add map layers as shown below.

Note that Leaflet requires you to add the JavaScript in the <body> element after the <div> map element.

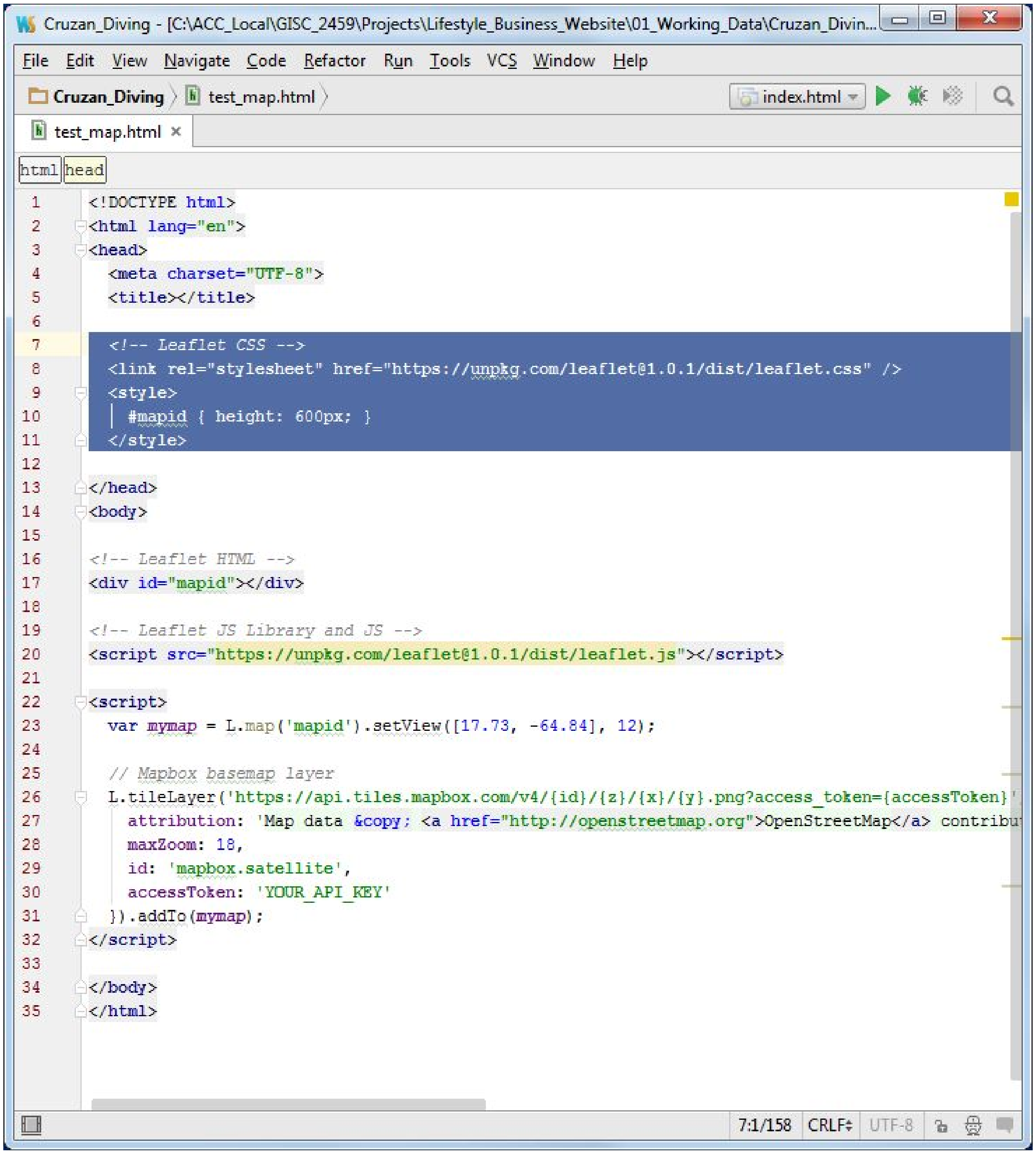

Cruzan_Diving - [C:\ACC_Local\GISC_2459\Projects\Lifestyle_Business_Website\01_Working_Data\Cruzan_Divin...
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Cruzan_Diving
test_map.html
index.html
html body
12:1/631 CRLF UTF-8

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title></title>
</head>
<body>

<!-- Leaflet HTML -->
<div id="mapid"></div>

<!-- Leaflet JS Library and JS -->
<script src="https://unpkg.com/leaflet@1.0.1/dist/leaflet.js"></script>

<script>
  var mymap = L.map('mapid').setView([17.73, -64.84], 12);

  // Mapbox basemap layer
  L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?access_token={accessToken}'
    attribution: 'Map data &copy; <a href="http://openstreetmap.org">OpenStreetMap</a> contribu
    maxZoom: 18,
    id: 'mapbox.satellite',
    accessToken: 'YOUR_API_KEY'
  }).addTo(mymap);
</script>

</body>
</html>
```

test_map.html with JavaScript Leaflet file, variable declarations, and initialized map

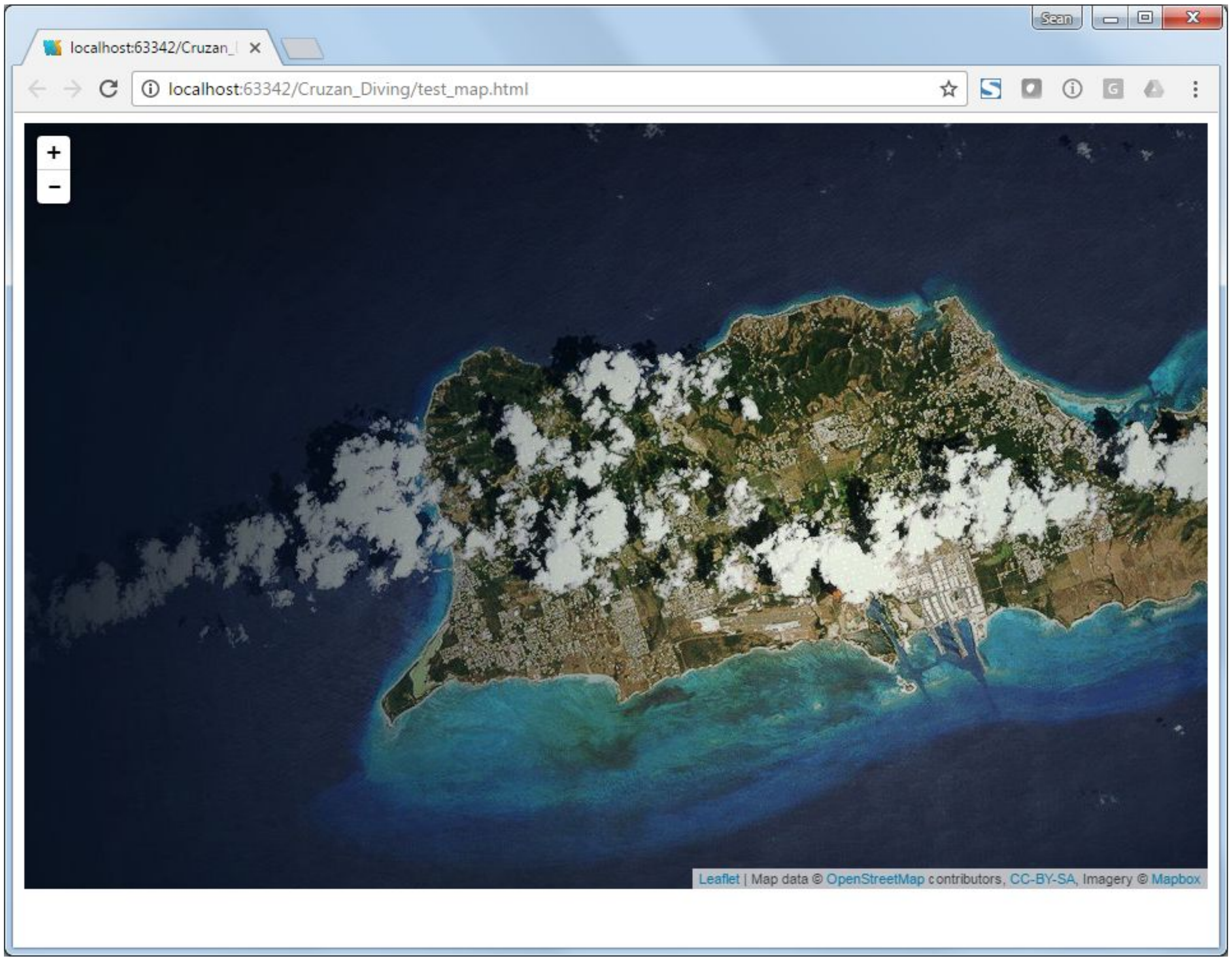
Add the CSS styling

Leaflet requires the Leaflet CSS file and CSS <style> that sets the map container height. The height of your map container will depend on where your map is embedded in your website. Add the CSS leaflet.css and #mapid styling for the HTML <div id="mapid"> as shown below.


Cruzan_Diving - [C:\ACC_Local\GISC_2459\Projects\Lifestyle_Business_Website\01_Working_Data\Cruzan_Divin...
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Cruzan_Diving
test_map.html
index.html
test_map.html
html head
7:1/158 CRLF UTF-8

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title></title>

  <!-- Leaflet CSS -->
  <link rel="stylesheet" href="https://unpkg.com/leaflet@1.0.1/dist/leaflet.css" />
  <style>
    #mapid { height: 600px; }
  </style>

</head>
<body>

<!-- Leaflet HTML -->
<div id="mapid"></div>

<!-- Leaflet JS Library and JS -->
<script src="https://unpkg.com/leaflet@1.0.1/dist/leaflet.js"></script>

<script>
  var mymap = L.map('mapid').setView([17.73, -64.84], 12);

  // Mapbox basemap layer
  L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?access_token={accessToken}'
    attribution: 'Map data &copy; <a href="http://openstreetmap.org">OpenStreetMap</a> contribu
    maxZoom: 18,
    id: 'mapbox.satellite',
    accessToken: 'YOUR_API_KEY'
  }).addTo(mymap);
</script>

</body>
</html>
```

test_map.html with CSS Leaflet file and CSS <style> that sets the map container height


localhost:63342/Cruzan_
localhost:63342/Cruzan_Diving/test_map.html
Leaflet | Map data © OpenStreetMap contributors, CC-BY-SA, Imagery © Mapbox

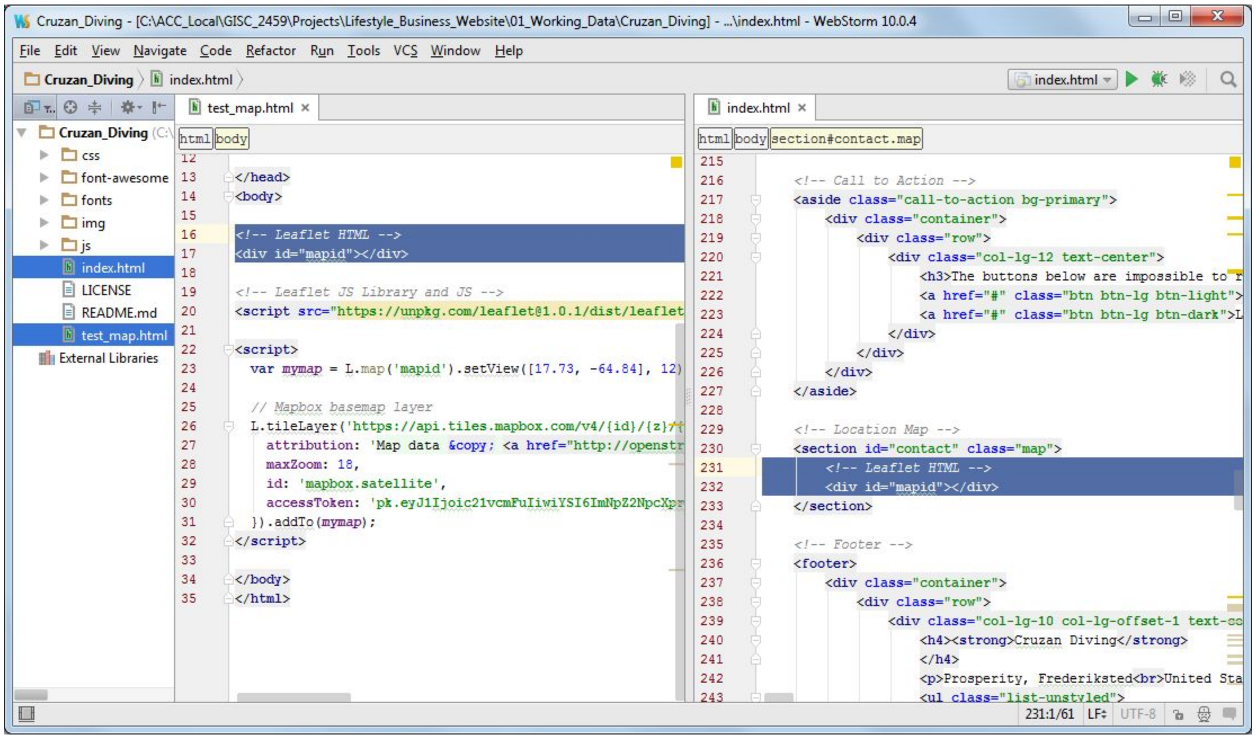
test_map.html with Leaflet interactive map viewed in an Internet browser

Add the Leaflet map to website

In Summary Task 1 Create Website, you created a website with an index page along with a spot reserved for your embedded interactive map. The next step is add the Leaflet Map API to your website.

The CSS (e.g. <style>), JavaScript (e.g. <script>), and HTML (e.g. <div>) elements found in test_map.html are organized into two main sections: the header (i.e. <head>) and the body (i.e. <body>). Copy the HTML, JavaScript, and CSS code from test_map.html to index.html and confirm that the map works as shown in the following images.

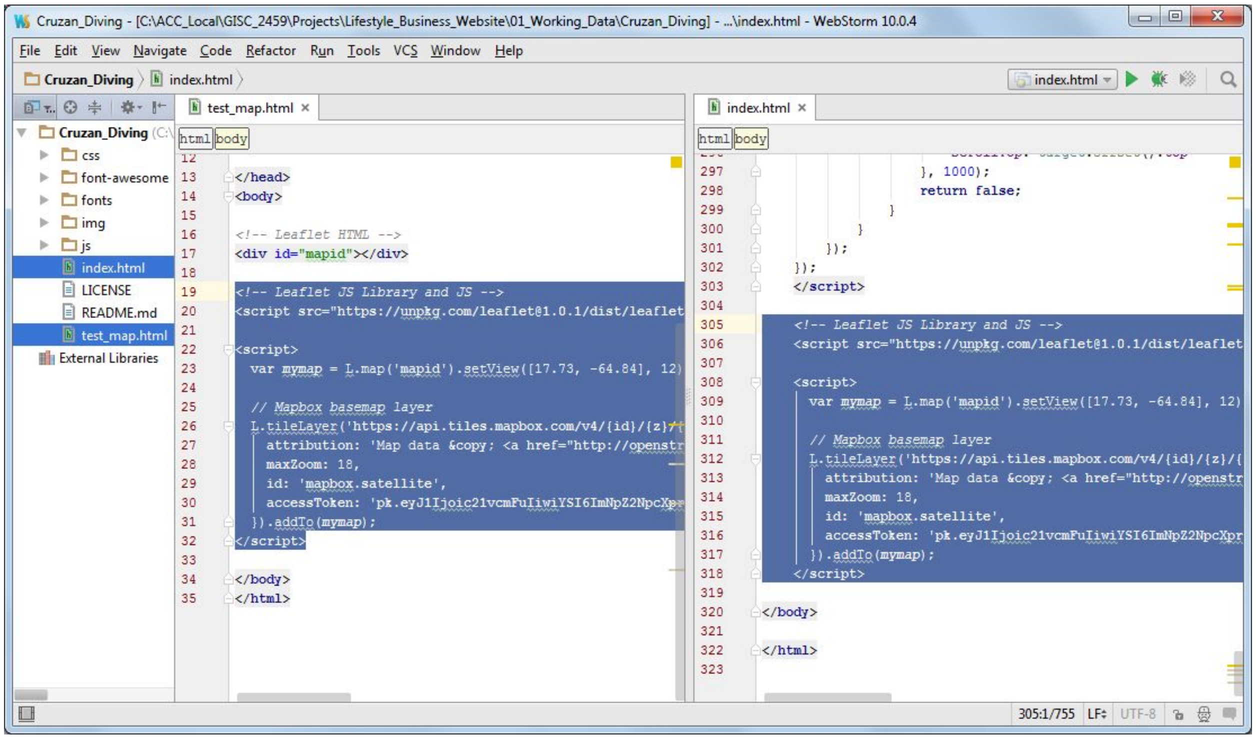
Remember the Leaflet API requires that you add the JavaScript in the <body> element after the <div> map element.


Cruzan_Diving - [C:\ACC_Local\GISC_2459\Projects\Lifestyle_Business_Website\01_Working_Data\Cruzan_Diving] - ...\index.html - WebStorm 10.0.4
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Cruzan_Diving
index.html
Cruzan_Diving (C:\
css
font-awesome
fonts
img
js
index.html
LICENSE
README.md
test_map.html
External Libraries
test_map.html
index.html
html body
html body section#contact.map
231:1/61 LF UTF-8

```
</head>
<body>

<!-- Leaflet HTML -->
<div id="mapid"></div>

<!-- Leaflet JS Library and JS -->
<script src="https://unpkg.com/leaflet@1.0.1/dist/leaflet

<script>
  var mymap = L.map('mapid').setView([17.73, -64.84], 12)

  // Mapbox basemap layer
  L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{
    attribution: 'Map data &copy; <a href="http://openstr
    maxZoom: 18,
    id: 'mapbox.satellite',
    accessToken: 'pk.eyJ1Ijoic21vcmFuIiwiYSI6ImNpZ2NpcXpr
  }).addTo(mymap);
</script>

</body>
</html>
```

```

    <!-- Call to Action -->
    <aside class="call-to-action bg-primary">
        <div class="container">
            <div class="row">
                <div class="col-lg-12 text-center">
                    <h3>The buttons below are impossible to r
                    <a href="#" class="btn btn-lg btn-light">
                    <a href="#" class="btn btn-lg btn-dark">L
                </div>
            </div>
        </div>
    </aside>

    <!-- Location Map -->
    <section id="contact" class="map">
        <!-- Leaflet HTML -->
        <div id="mapid"></div>
    </section>

    <!-- Footer -->
    <footer>
        <div class="container">
            <div class="row">
                <div class="col-lg-10 col-lg-offset-1 text-ce
                    <h4><strong>Cruzan Diving</strong>
                    </h4>
                    <p>Prosperity, Frederiksted<br>United Sta
                    <ul class="list-unstyled">
```

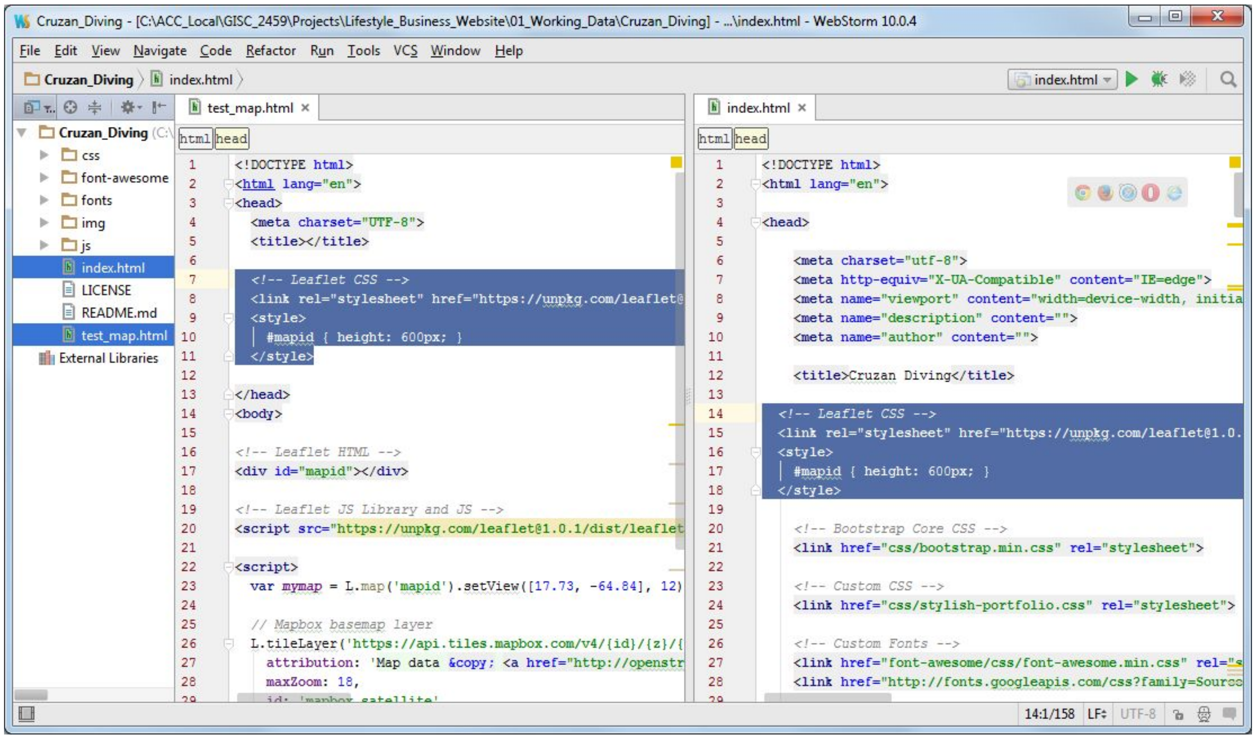
Copy the HTML <div> code from test_map.html to index.html


Cruzan_Diving - [C:\ACC_Local\GISC_2459\Projects\Lifestyle_Business_Website\01_Working_Data\Cruzan_Diving] - ...\index.html - WebStorm 10.0.4
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Cruzan_Diving
index.html
Cruzan_Diving (C:\
css
font-awesome
fonts
img
js
index.html
LICENSE
README.md
test_map.html
External Libraries
test_map.html
index.html
html body
305:1/755 LF UTF-8

```
</head>
<body>

<!-- Leaflet HTML -->
<div id="mapid"></div>

<!-- Leaflet JS Library and JS -->
<script src="https://unpkg.com/leaflet@1.0.1/dist/leaflet

<script>
  var mymap = L.map('mapid').setView([17.73, -64.84], 12)

  // Mapbox basemap layer
  L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{
    attribution: 'Map data &copy; <a href="http://openstr
    maxZoom: 18,
    id: 'mapbox.satellite',
    accessToken: 'pk.eyJ1Ijoic21vcmFuIiwiYSI6ImNpZ2NpcXpr
  }).addTo(mymap);
</script>

</body>
</html>
```

```
                    }, 1000);
                    return false;
                }
            }
        });
    });
    </script>

    <!-- Leaflet JS Library and JS -->
    <script src="https://unpkg.com/leaflet@1.0.1/dist/leaflet

    <script>
      var mymap = L.map('mapid').setView([17.73, -64.84], 12)

      // Mapbox basemap layer
      L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{
        attribution: 'Map data &copy; <a href="http://openstr
        maxZoom: 18,
        id: 'mapbox.satellite',
        accessToken: 'pk.eyJ1Ijoic21vcmFuIiwiYSI6ImNpZ2NpcXpr
      }).addTo(mymap);
    </script>

</body>

</html>

```

Copy the JavaScript <script> code from test_map.html to index.html



Cruzan_Diving - [C:\ACC_Local\GISC_2459\Projects\Lifestyle_Business_Website\01_Working_Data\Cruzan_Diving] - ...\index.html - WebStorm 10.0.4

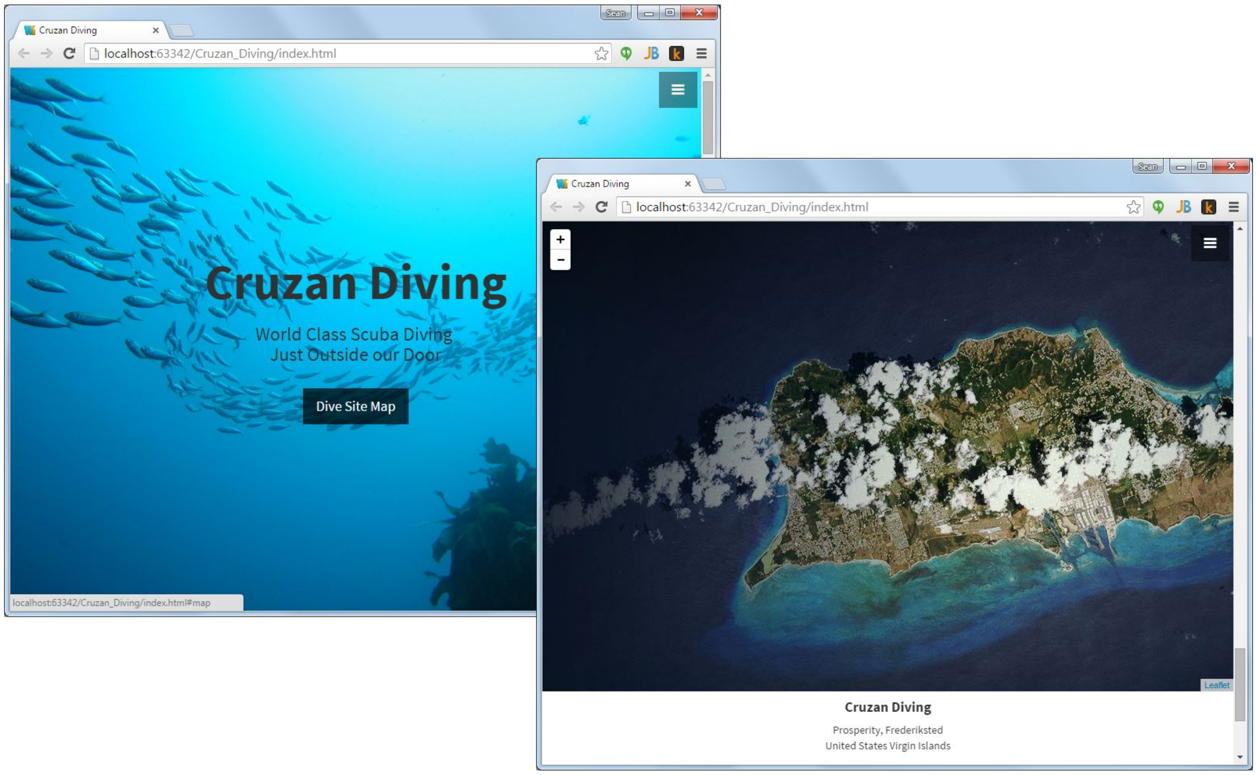
test_map.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title></title>

  <!-- Leaflet CSS -->
  <link rel="stylesheet" href="https://unpkg.com/leaflet@
  <style>
    #mapid { height: 600px; }
  </style>

</head>
<body>

<!-- Leaflet HTML -->
<div id="mapid"></div>

<!-- Leaflet JS Library and JS -->
<script src="https://unpkg.com/leaflet@1.0.1/dist/leaflet

<script>
  var mymap = L.map('mapid').setView([17.73, -64.84], 12)

  // Mapbox basemap layer
  L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{
    attribution: 'Map data &copy; <a href="http://openstr
    maxZoom: 18,
```

index.html

```
<!DOCTYPE html>
<html lang="en">

<head>

    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initia
    <meta name="description" content="">
    <meta name="author" content="">

    <title>Cruzan Diving</title>

  <!-- Leaflet CSS -->
  <link rel="stylesheet" href="https://unpkg.com/leaflet@1.0.
  <style>
    #mapid { height: 600px; }
  </style>

    <!-- Bootstrap Core CSS -->
    <link href="css/bootstrap.min.css" rel="stylesheet">

    <!-- Custom CSS -->
    <link href="css/stylish-portfolio.css" rel="stylesheet">

    <!-- Custom Fonts -->
    <link href="font-awesome/css/font-awesome.min.css" rel="s
    <link href="http://fonts.googleapis.com/css?family=Sourc
```

Copy the CSS <style> code from test_map.html to index.html



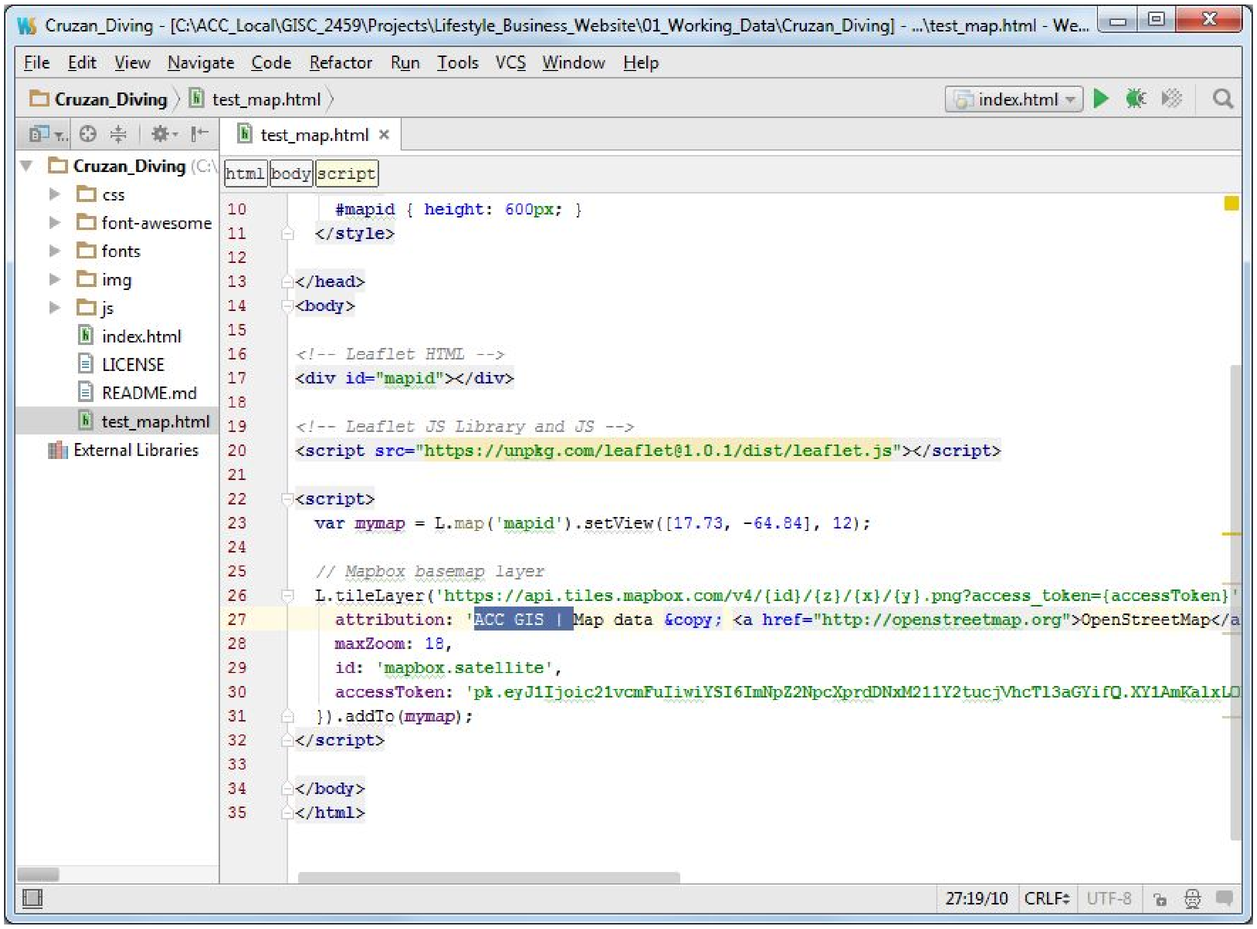
The lifestyle business website with embedded interactive map

4. Add Layers to Leaflet Map

Your website now includes a successfully embedded Leaflet interactive map that displays a Mapbox cached map service. The Leaflet API also includes an attribution property that allows you to credit data sources like Mapbox. In Summary Task 4, you'll learn how to update the Leaflet API attribution property; add your business location, nearby attractions, and additional basemap layers; and add a toggle layers control.

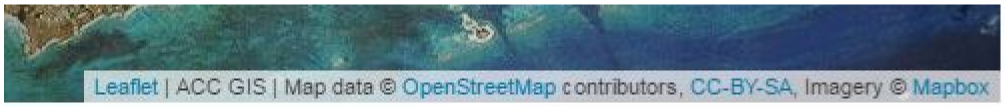
Update Leaflet attribution property

Look in the lower, right-hand corner of your embedded Leaflet interactive map. Notice that the text "Leaflet" is displayed by default along with a hyperlink to the Leaflet API website. This text and link can be updated using the Leaflet attribution property as shown below.


Cruzan_Diving - [C:\ACC_Local\GISC_2459\Projects\Lifestyle_Business_Website\01_Working_Data\Cruzan_Diving] - ...\test_map.html - We...
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Cruzan_Diving test_map.html
index.html
test_map.html
Cruzan_Diving
css
font-awesome
fonts
img
js
index.html
LICENSE
README.md
test_map.html
External Libraries
html body script
27:19/10 CRLF UTF-8

```
    #mapid { height: 600px; }
  </style>

</head>
<body>

<!-- Leaflet HTML -->
<div id="mapid"></div>

<!-- Leaflet JS Library and JS -->
<script src="https://unpkg.com/leaflet@1.0.1/dist/leaflet.js"></script>

<script>
  var mymap = L.map('mapid').setView([17.73, -64.84], 12);

  // Mapbox basemap layer
  L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?access_token={accessToken}'
    attribution: 'ACC GIS | Map data &copy; <a href="http://openstreetmap.org">OpenStreetMap</a
    maxZoom: 18,
    id: 'mapbox.satellite',
    accessToken: 'pk.eyJ1Ijoic21vcmFuIiwiYSI6ImNpZ2NpcXprdDNxM211Y2tucjVhcTl3aGYifQ.XY1AmKalxLD
  }).addTo(mymap);
</script>

</body>
</html>
```

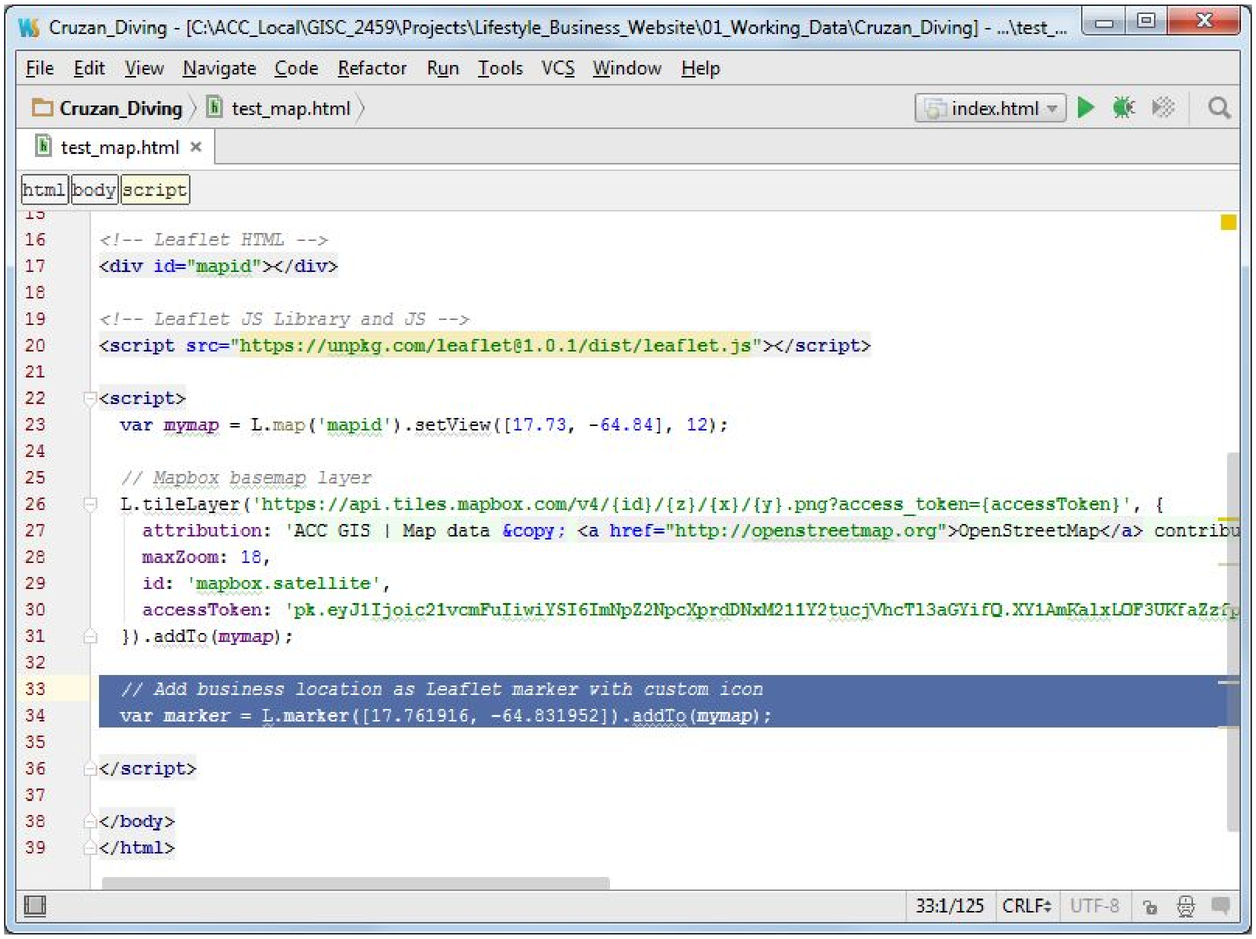
test_html with the updated Leaflet attribution property crediting ACC GIS


Leaflet | ACC GIS | Map data © OpenStreetMap contributors, CC-BY-SA, Imagery © Mapbox

The updated attribution property found in the lower, right-hand corner of the map

Add Leaflet marker with custom icon and popup

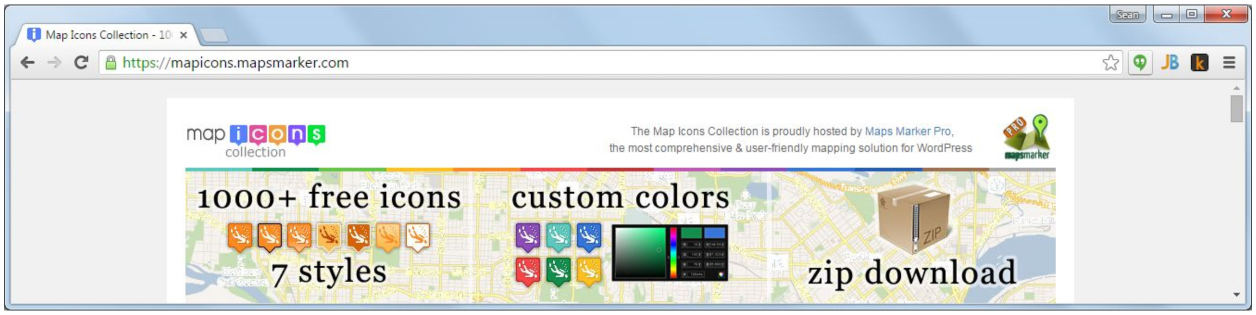
It's easy to add points, lines, and polygonal graphics to a Leaflet map. Using the Leaflet Quick Start Guide found in the Leaflet Tutorials as a guide, add your business location's latitude and longitude coordinates as a marker to test_map.html as shown below.


Cruzan_Diving - [C:\ACC_Local\GISC_2459\Projects\Lifestyle_Business_Website\01_Working_Data\Cruzan_Diving] - ...\test_...
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Cruzan_Diving
test_map.html
index.html
html body script
33:1/125 CRLF UTF-8

```
<!-- Leaflet HTML -->
<div id="mapid"></div>

<!-- Leaflet JS Library and JS -->
<script src="https://unpkg.com/leaflet@1.0.1/dist/leaflet.js"></script>

<script>
  var mymap = L.map('mapid').setView([17.73, -64.84], 12);

  // Mapbox basemap layer
  L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?access_token={accessToken}', {
    attribution: 'ACC GIS | Map data &copy; <a href="http://openstreetmap.org">OpenStreetMap</a> contribu
    maxZoom: 18,
    id: 'mapbox.satellite',
    accessToken: 'pk.eyJ1Ijoic21vcmFuIiwiYSI6ImNpZ2NpcXprdDNxM211Y2tucjVhcTl3aGYifQ.XY1AmKalxLOF3UKfaZzf
  }).addTo(mymap);

  // Add business location as Leaflet marker with custom icon
  var marker = L.marker([17.761916, -64.831952]).addTo(mymap);

</script>

</body>
</html>
```

test_map.html with marker added to the map

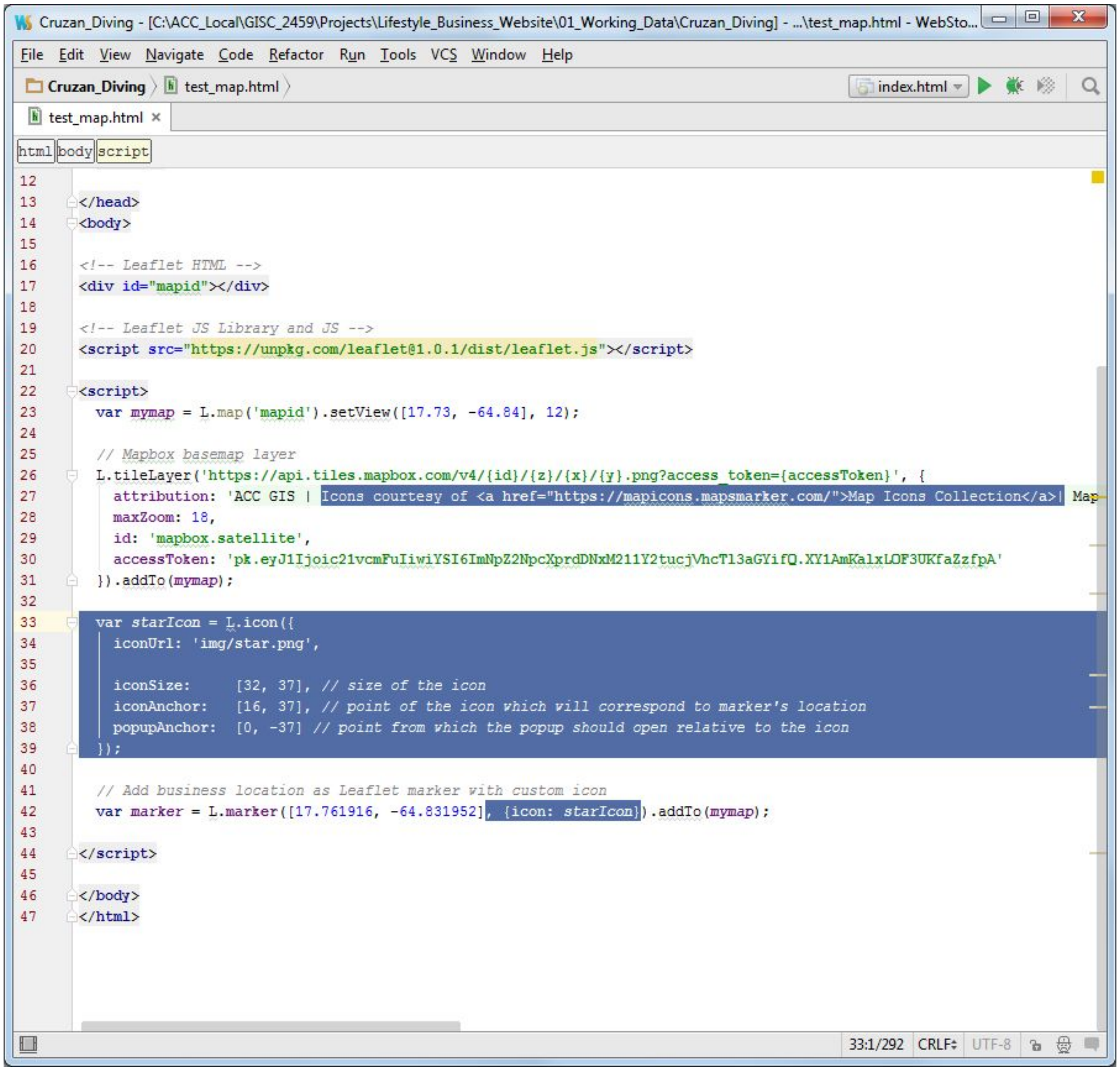
The Leaflet default marker icon looks good, but a custom icon like one from Map Icons will look great. Find a custom icon you like and save it to your website folder in the img or images folder.


Map Icons Collection - 10
https://mapicons.mapsmarker.com
map icons collection
The Map Icons Collection is proudly hosted by Maps Marker Pro, the most comprehensive & user-friendly mapping solution for WordPress
1000+ free icons
7 styles
custom colors
zip download

Map Icons is a set of free and customizable map icons made available via the Creative Commons License

Using the Leaflet Markers with Custom Icons Tutorial as a guide, add the L.icon object and properties to test_map.html. Be sure to use iconSize, iconAnchor, and popupAnchor properties that are consistent with the custom icon you've selected. Now, add the the icon property to the L.marker object.

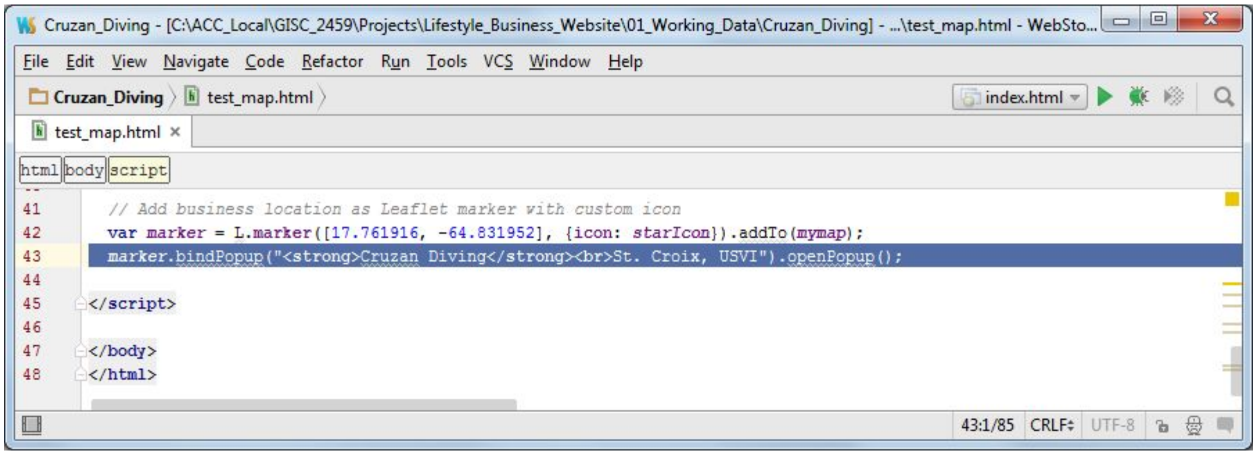
Map Icons requests to be credited as well, so be sure to update the attribution property if needed.


Cruzan_Diving - [C:\ACC_Local\GISC_2459\Projects\Lifestyle_Business_Website\01_Working_Data\Cruzan_Diving] - ...\test_map.html - WebSto...
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Cruzan_Diving
test_map.html
index.html
html body script
33:1/292 CRLF UTF-8

```
</head>
<body>

<!-- Leaflet HTML -->
<div id="mapid"></div>

<!-- Leaflet JS Library and JS -->
<script src="https://unpkg.com/leaflet@1.0.1/dist/leaflet.js"></script>

<script>
  var mymap = L.map('mapid').setView([17.73, -64.84], 12);

  // Mapbox basemap layer
  L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?access_token={accessToken}', {
    attribution: 'ACC GIS | Icons courtesy of <a href="https://mapicons.mapsmarker.com/">Map Icons Collection</a>| Map
    maxZoom: 18,
    id: 'mapbox.satellite',
    accessToken: 'pk.eyJ1Ijoic21vcmFuIiwiYSI6ImNpZ2NpcXprdDNxM211Y2tucjVhcTl3aGYifQ.XY1AmKalxLOF3UKfaZzfpA'
  }).addTo(mymap);

  var starIcon = L.icon({
    iconUrl: 'img/star.png',

    iconSize:     [32, 37], // size of the icon
    iconAnchor:   [16, 37], // point of the icon which will correspond to marker's location
    popupAnchor:  [0, -37] // point from which the popup should open relative to the icon
  });

  // Add business location as Leaflet marker with custom icon
  var marker = L.marker([17.761916, -64.831952], {icon: starIcon}).addTo(mymap);

</script>

</body>
</html>
```

test_map.html with marker symbolized with a custom icon

Lastly, bind an open popup to the marker by updating test_map.html as shown below.

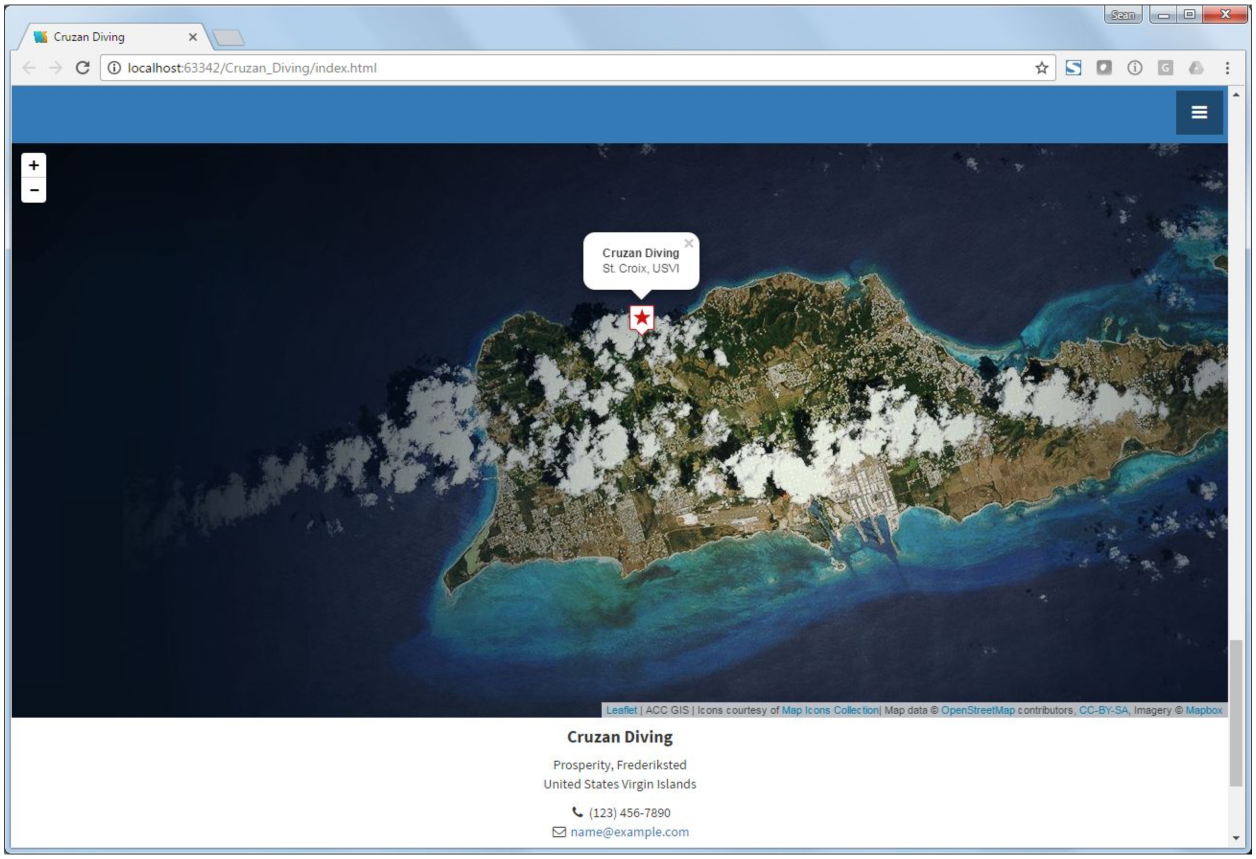


Cruzan_Diving - [C:\ACC_Local\GISC_2459\Projects\Lifestyle_Business_Website\01_Working_Data\Cruzan_Diving] - ...\test_map.html - WebSto...

```
    // Add business location as Leaflet marker with custom icon
    var marker = L.marker([17.761916, -64.831952], {icon: starIcon}).addTo(mymap);
    marker.bindPopup("<strong>Cruzan Diving</strong><br>St. Croix, USVI").openPopup();

</script>

</body>
</html>
```

test_map.html with open popup window bound to marker

Now copy the updated JavaScript code from test_map.html to index.html and view the updated website and map in a browser. Your map should now include the location of your business with custom icon and a popup window that is open when the map loads.



localhost:63342/Cruzan_Diving/index.html

Cruzan Diving
St. Croix, USVI

Leaflet | ACC GIS | Icons courtesy of Map Icons Collection| Map data © OpenStreetMap contributors, CC-BY-SA, Imagery © Mapbox

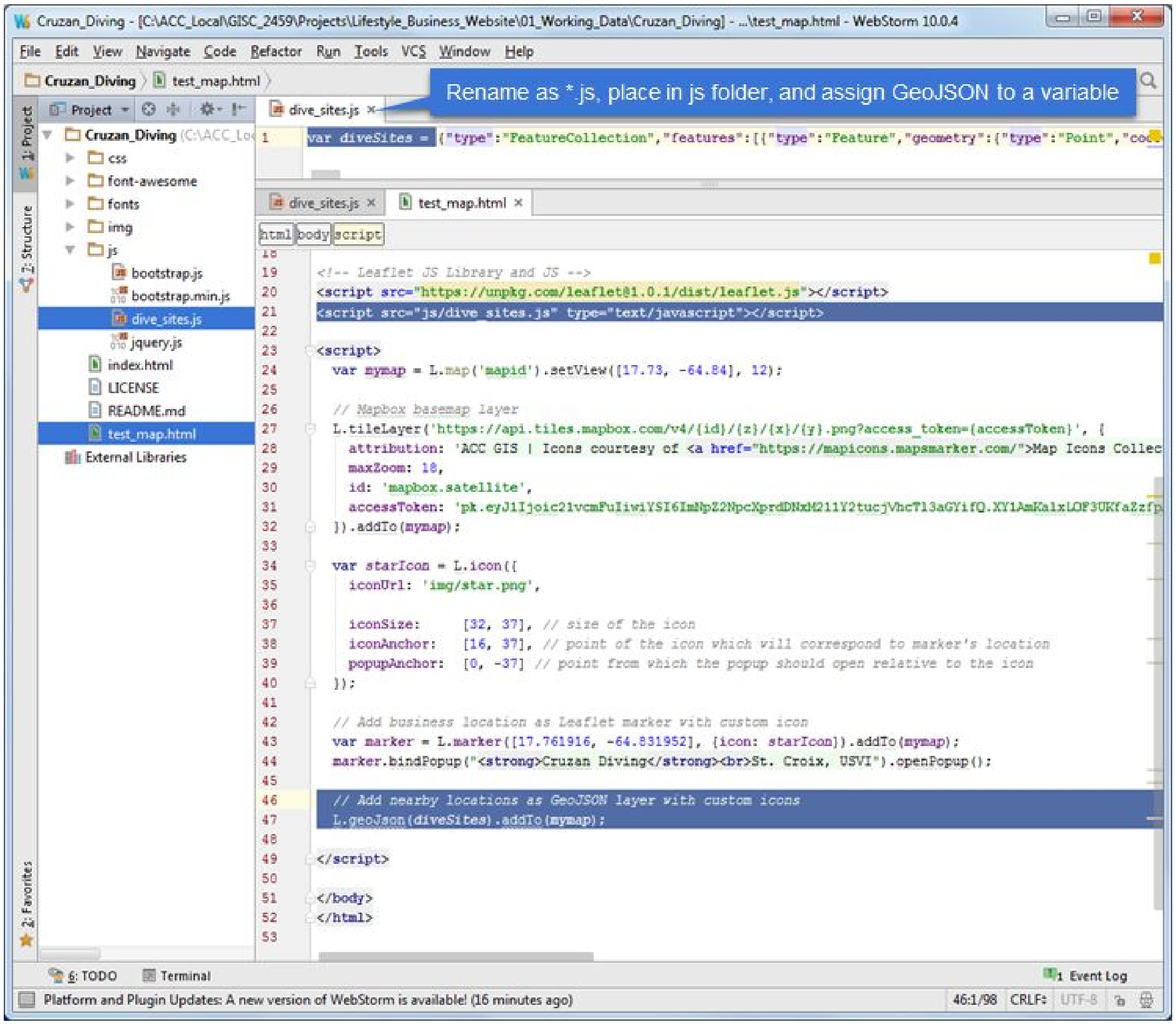
Cruzan Diving
Prosperity, Frederiksted
United States Virgin Islands
(123) 456-7890
name@example.com

index.html with Leaflet marker with custom icon and popup; and updated attribution

Add GeoJSON features with custom icons and popup windows

In addition to your business location, you want your map to include nearby attractions. You can add the GeoJSON features you created during Summary Task 2 Collect and Assimilate Data directly into your Leaflet interactive map.

Using the Leaflet GeoJSON with Leaflet Tutorial as a general guide and the How to load and external GeoJSON file into Leaflet map StackExchange post, add the nearby attractions GeoJSON file to test_map.html as shown below.



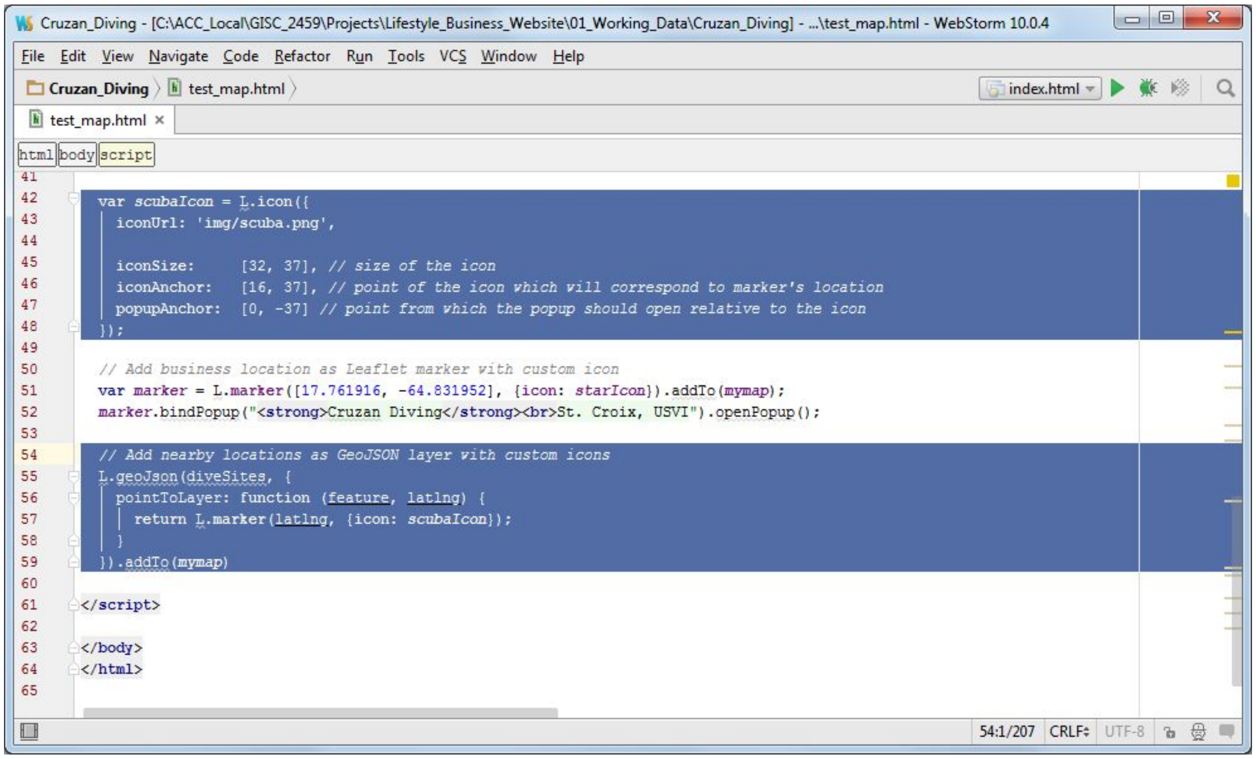
```
var diveSites = {"type":"FeatureCollection","features":[{"type":"Feature","geometry":{"type":"Point","coo
```

```
<!-- Leaflet JS Library and JS -->
<script src="https://unpkg.com/leaflet@1.0.1/dist/leaflet.js"></script>
<script src="js/dive_sites.js" type="text/javascript"></script>

<script>
  var mymap = L.map('mapid').setView([17.73, -64.84], 12);

  // Mapbox basemap layer
  L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?access_token={accessToken}', {
    attribution: 'ACC GIS | Icons courtesy of <a href="https://mapicons.mapsmarker.com/">Map Icons Collec
    maxZoom: 18,
    id: 'mapbox.satellite',
    accessToken: 'pk.eyJ1Ijoic21vcmFuIiwiYSI6ImNpZ2NpcXprdDNxM211Y2tucjVhcTl3aGYifQ.XY1AmKalxLOF3UKfaZzfp
  }).addTo(mymap);

  var starIcon = L.icon({
    iconUrl: 'img/star.png',

    iconSize:     [32, 37], // size of the icon
    iconAnchor:   [16, 37], // point of the icon which will correspond to marker's location
    popupAnchor:  [0, -37] // point from which the popup should open relative to the icon
  });

  // Add business location as Leaflet marker with custom icon
  var marker = L.marker([17.761916, -64.831952], {icon: starIcon}).addTo(mymap);
  marker.bindPopup("<strong>Cruzan Diving</strong><br>St. Croix, USVI").openPopup();

  // Add nearby locations as GeoJSON layer with custom icons
  L.geoJson(diveSites).addTo(mymap);

</script>

</body>
</html>
```

Platform and Plugin Updates: A new version of WebStorm is available! (16 minutes ago)

test_map.html with updated GeoJSON file extension and code (see top panel above) added to the map

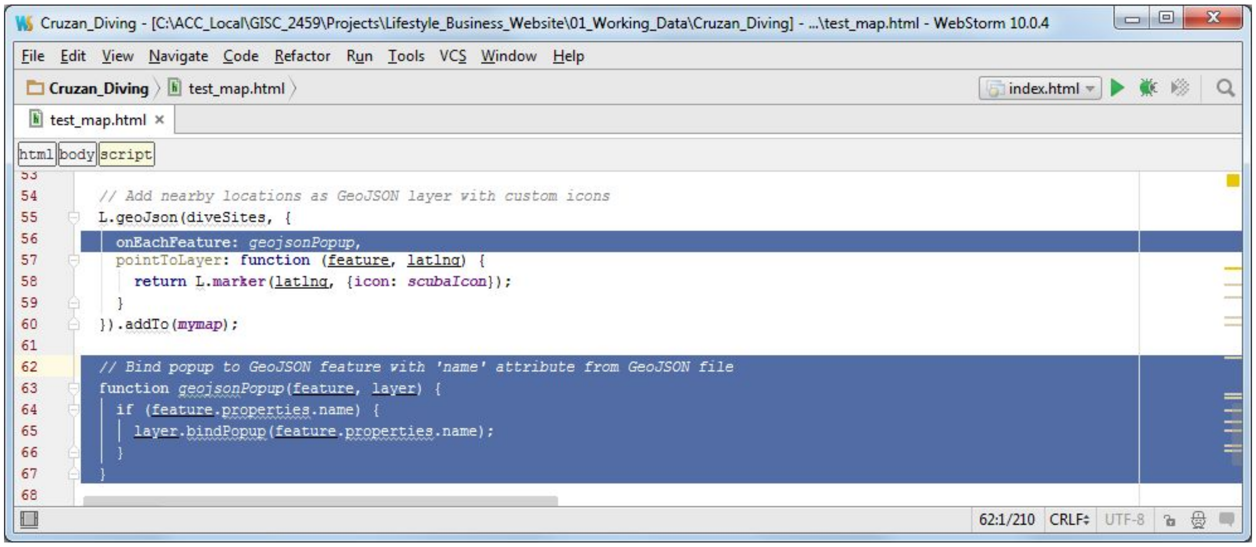
View test_map.html in a browser. You should see your nearby attractions mapped with the default marker icon. Mapping GeoJSON point features in Leaflet with custom icons requires that you load the GeoJSON point features into a Leaflet layer first. You can then update the icon property to include a custom icon as shown on the following page.


Cruzan_Diving - [C:\ACC_Local\GISC_2459\Projects\Lifestyle_Business_Website\01_Working_Data\Cruzan_Diving] - ...\test_map.html - WebStorm 10.0.4
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Cruzan_Diving test_map.html index.html
html body script
54:1/207 CRLF UTF-8

```
    var scubaIcon = L.icon({
      iconUrl: 'img/scuba.png',

      iconSize:     [32, 37], // size of the icon
      iconAnchor:   [16, 37], // point of the icon which will correspond to marker's location
      popupAnchor:  [0, -37] // point from which the popup should open relative to the icon
    });

    // Add business location as Leaflet marker with custom icon
    var marker = L.marker([17.761916, -64.831952], {icon: starIcon}).addTo(mymap);
    marker.bindPopup("<strong>Cruzan Diving</strong><br>St. Croix, USVI").openPopup();

    // Add nearby locations as GeoJSON layer with custom icons
    L.geoJson(diveSites, {
      pointToLayer: function (feature, latlng) {
        return L.marker(latlng, {icon: scubaIcon});
      }
    }).addTo(mymap)

  </script>

  </body>
  </html>
```

test_map.html with GeoJSON markers symbolized with a custom icon

Lastly, bind popups to the newly created Leaflet layer by adding a geojsonPopup function that opens a popup window when each marker is clicked (i.e. the onEachFeature event property).


Cruzan_Diving - [C:\ACC_Local\GISC_2459\Projects\Lifestyle_Business_Website\01_Working_Data\Cruzan_Diving] - ...\test_map.html - WebStorm 10.0.4
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Cruzan_Diving test_map.html index.html
html body script
62:1/210 CRLF UTF-8

```
    // Add nearby locations as GeoJSON layer with custom icons
    L.geoJson(diveSites, {
      onEachFeature: geojsonPopup,
      pointToLayer: function (feature, latlng) {
        return L.marker(latlng, {icon: scubaIcon});
      }
    }).addTo(mymap);

    // Bind popup to GeoJSON feature with 'name' attribute from GeoJSON file
    function geojsonPopup(feature, layer) {
      if (feature.properties.name) {
        layer.bindPopup(feature.properties.name);
      }
    }
```

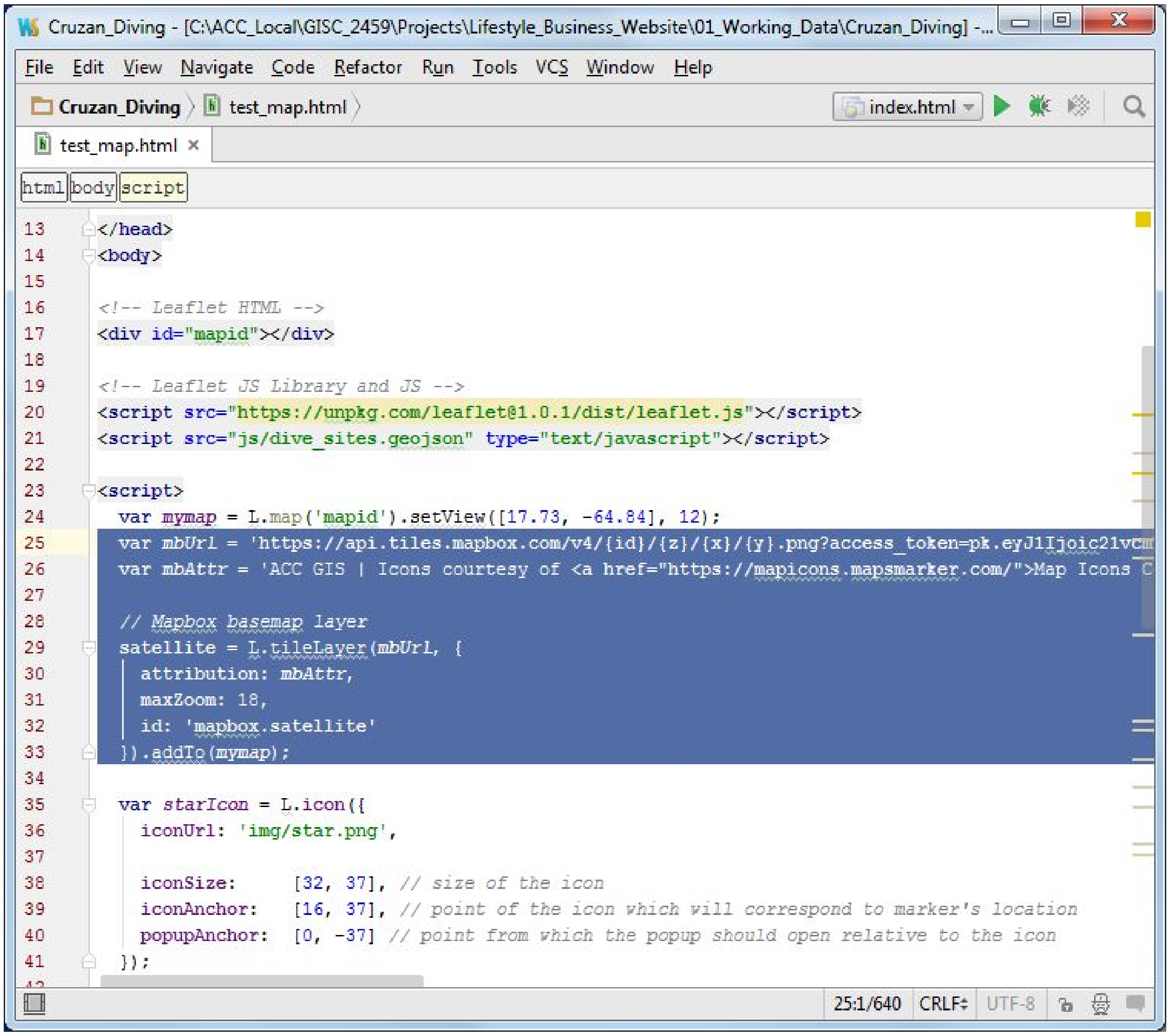
test_map.html with popup window function called when GeoJSON markers are clicked

Now copy the updated JavaScript code from test_map.html to index.html and view the updated map in a browser. Your map should now include your nearby attractions with custom icons.

Add Additional Basemap Layers and Layer Toggle Control

The last step is to add additional basemap layers and a layer toggle control. This will allow website visitors to toggle between selected Mapbox cached map services, as well as toggle the nearby attractions layer on and off.

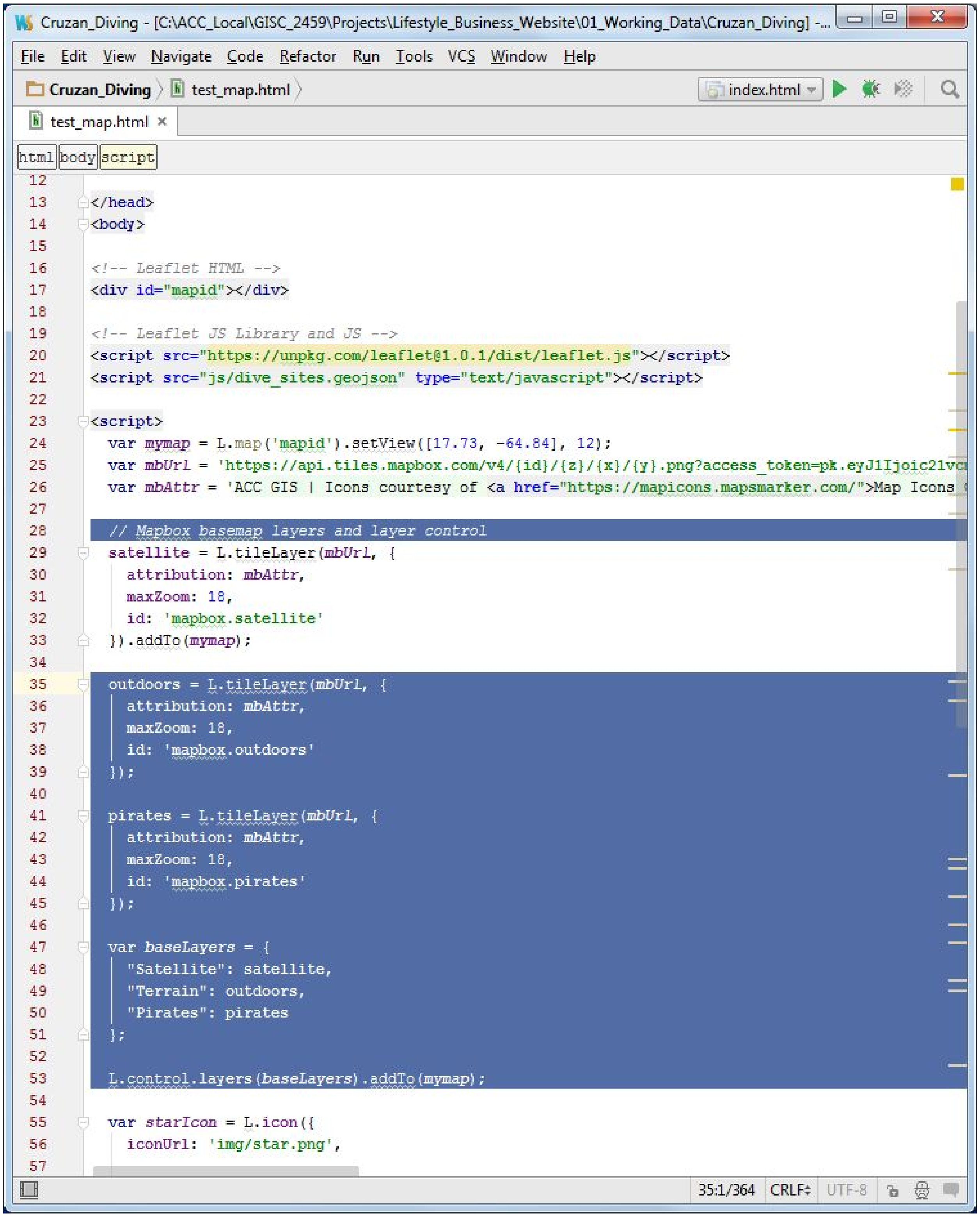
Using the Leaflet Layer Groups and Layers Control Tutorial as a general guide, rearrange your L.tileLayer code as shown below. This will allow you to add additional layers without repeating the Mapbox tile URL and attribution. Confirm that test_map.html still works as expected before proceeding.


Cruzan_Diving - [C:\ACC_Local\GISC_2459\Projects\Lifestyle_Business_Website\01_Working_Data\Cruzan_Diving] -...
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Cruzan_Diving
test_map.html
index.html
test_map.html
html body script
25:1/640 CRLF UTF-8

```
</head>
<body>

<!-- Leaflet HTML -->
<div id="mapid"></div>

<!-- Leaflet JS Library and JS -->
<script src="https://unpkg.com/leaflet@1.0.1/dist/leaflet.js"></script>
<script src="js/dive_sites.geojson" type="text/javascript"></script>

<script>
  var mymap = L.map('mapid').setView([17.73, -64.84], 12);
  var mbUrl = 'https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?access_token=pk.eyJ1Ijoic21vcm
  var mbAttr = 'ACC GIS | Icons courtesy of <a href="https://mapicons.mapsmarker.com/">Map Icons C

  // Mapbox basemap layer
  satellite = L.tileLayer(mbUrl, {
    attribution: mbAttr,
    maxZoom: 18,
    id: 'mapbox.satellite'
  }).addTo(mymap);

  var starIcon = L.icon({
    iconUrl: 'img/star.png',

    iconSize:     [32, 37], // size of the icon
    iconAnchor:   [16, 37], // point of the icon which will correspond to marker's location
    popupAnchor:  [0, -37] // point from which the popup should open relative to the icon
  });
```

test_map.html with added Mapbox basemap layers

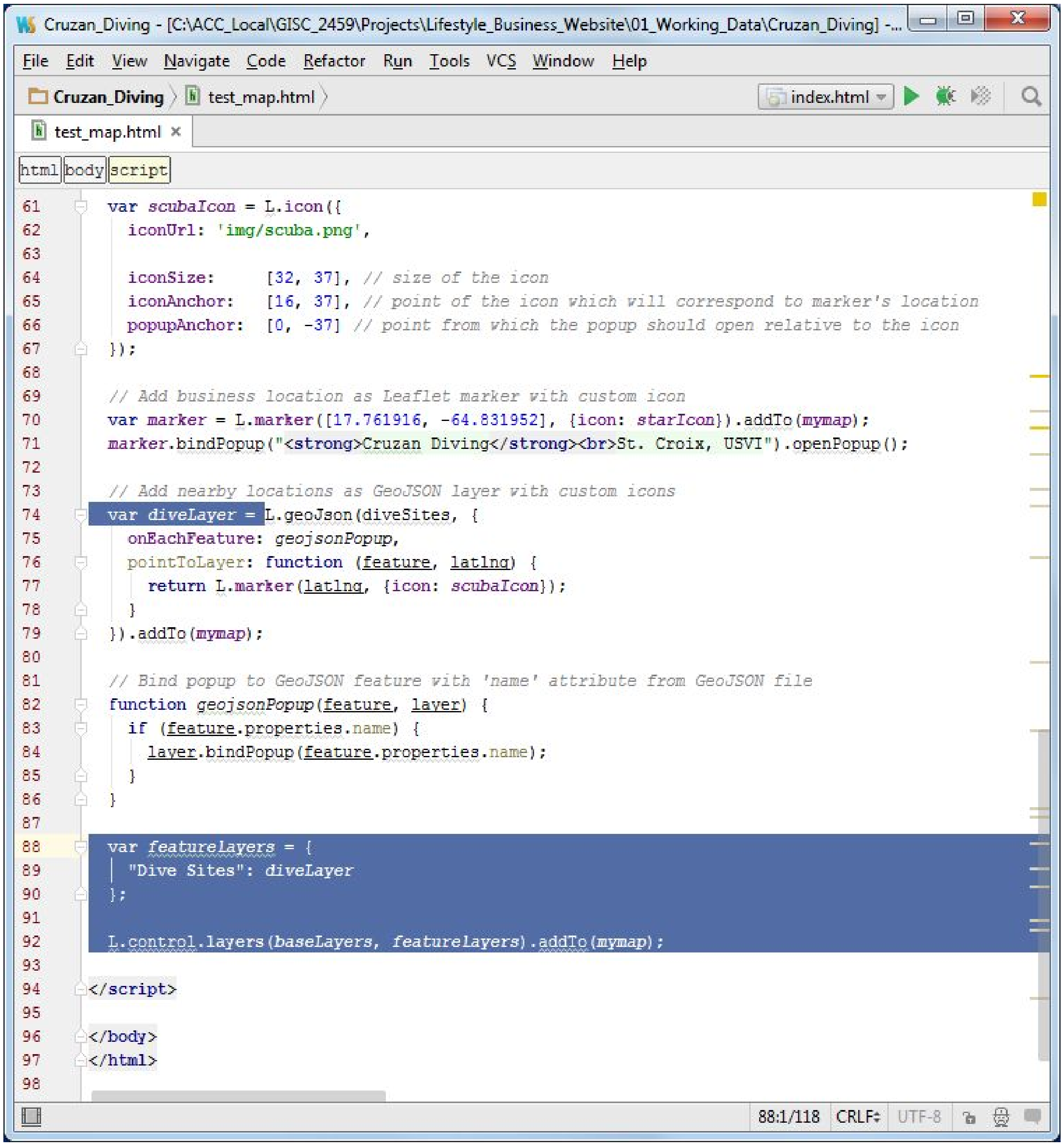
Now, review the Mapbox map services and select another basemap or two that will enhance your interactive map. Record the Map ID(s). Next, you'll add these additional Mapbox layers and a toggle control that toggles between them as shown below.


Cruzan_Diving - [C:\ACC_Local\GISC_2459\Projects\Lifestyle_Business_Website\01_Working_Data\Cruzan_Diving] -...
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Cruzan_Diving
test_map.html
index.html
test_map.html
html body script
35:1/364 CRLF UTF-8

```html
</head>
<body>

<!-- Leaflet HTML -->
<div id="mapid"></div>

<!-- Leaflet JS Library and JS -->
<script src="https://unpkg.com/leaflet@1.0.1/dist/leaflet.js"></script>
<script src="js/dive_sites.geojson" type="text/javascript"></script>

<script>
  var mymap = L.map('mapid').setView([17.73, -64.84], 12);
  var mbUrl = 'https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?access_token=pk.eyJ1Ijoic21vc
  var mbAttr = 'ACC GIS | Icons courtesy of <a href="https://mapicons.mapsmarker.com/">Map Icons

  // Mapbox basemap layers and layer control
  satellite = L.tileLayer(mbUrl, {
    attribution: mbAttr,
    maxZoom: 18,
    id: 'mapbox.satellite'
  }).addTo(mymap);

  outdoors = L.tileLayer(mbUrl, {
    attribution: mbAttr,
    maxZoom: 18,
    id: 'mapbox.outdoors'
  });

  pirates = L.tileLayer(mbUrl, {
    attribution: mbAttr,
    maxZoom: 18,
    id: 'mapbox.pirates'
  });

  var baseLayers = {
    "Satellite": satellite,
    "Terrain": outdoors,
    "Pirates": pirates
  };

  L.control.layers(baseLayers).addTo(mymap);

  var starIcon = L.icon({
    iconUrl: 'img/star.png',
```

test_map.html with additional basemap layers and toggle baseLayers control

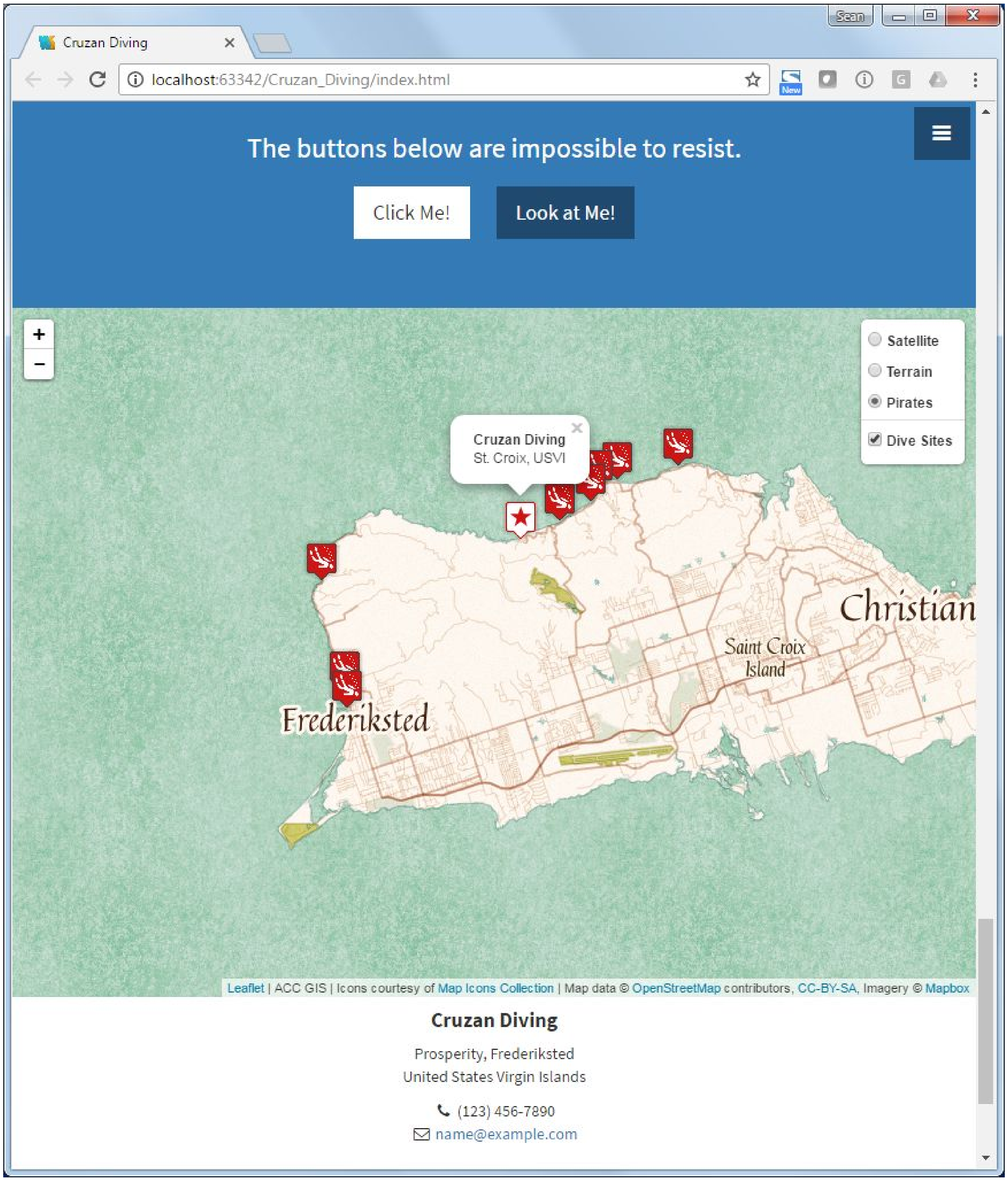
It's possible your nearby attractions might obscure something on the basemap that interests your user. You can add the nearby attractions GeoJSON layer to the toggle layer by assigning the L.geoJson code to a diveLayer and featureLayers variables and then adding the featureLayers variable to the L.control toggle as shown below. Note the updated L.control code is moved from just below the baseLayers variable to just below the featureLayers variable before the closing </script> tag.


Cruzan_Diving - [C:\ACC_Local\GISC_2459\Projects\Lifestyle_Business_Website\01_Working_Data\Cruzan_Diving] -...
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Cruzan_Diving
test_map.html
index.html
html body script
88:1/118 CRLF UTF-8

```
var scubaIcon = L.icon({
  iconUrl: 'img/scuba.png',

  iconSize:     [32, 37], // size of the icon
  iconAnchor:   [16, 37], // point of the icon which will correspond to marker's location
  popupAnchor:  [0, -37] // point from which the popup should open relative to the icon
});

// Add business location as Leaflet marker with custom icon
var marker = L.marker([17.761916, -64.831952], {icon: starIcon}).addTo(mymap);
marker.bindPopup("<strong>Cruzan Diving</strong><br>St. Croix, USVI").openPopup();

// Add nearby locations as GeoJSON layer with custom icons
var diveLayer = L.geoJson(diveSites, {
  onEachFeature: geojsonPopup,
  pointToLayer: function (feature, latlng) {
    return L.marker(latlng, {icon: scubaIcon});
  }
}).addTo(mymap);

// Bind popup to GeoJSON feature with 'name' attribute from GeoJSON file
function geojsonPopup(feature, layer) {
  if (feature.properties.name) {
    layer.bindPopup(feature.properties.name);
  }
}

var featureLayers = {
  "Dive Sites": diveLayer
};

L.control.layers(baseLayers, featureLayers).addTo(mymap);

</script>

</body>
</html>
```

test_map.html with updated baseLayers and featureLayers toggle

Confirm that test_map.html has a working basemap and features toggle before proceeding. Now copy the updated JavaScript code from test_map.html to index.html and view the updated map with toggle layer control in a browser.

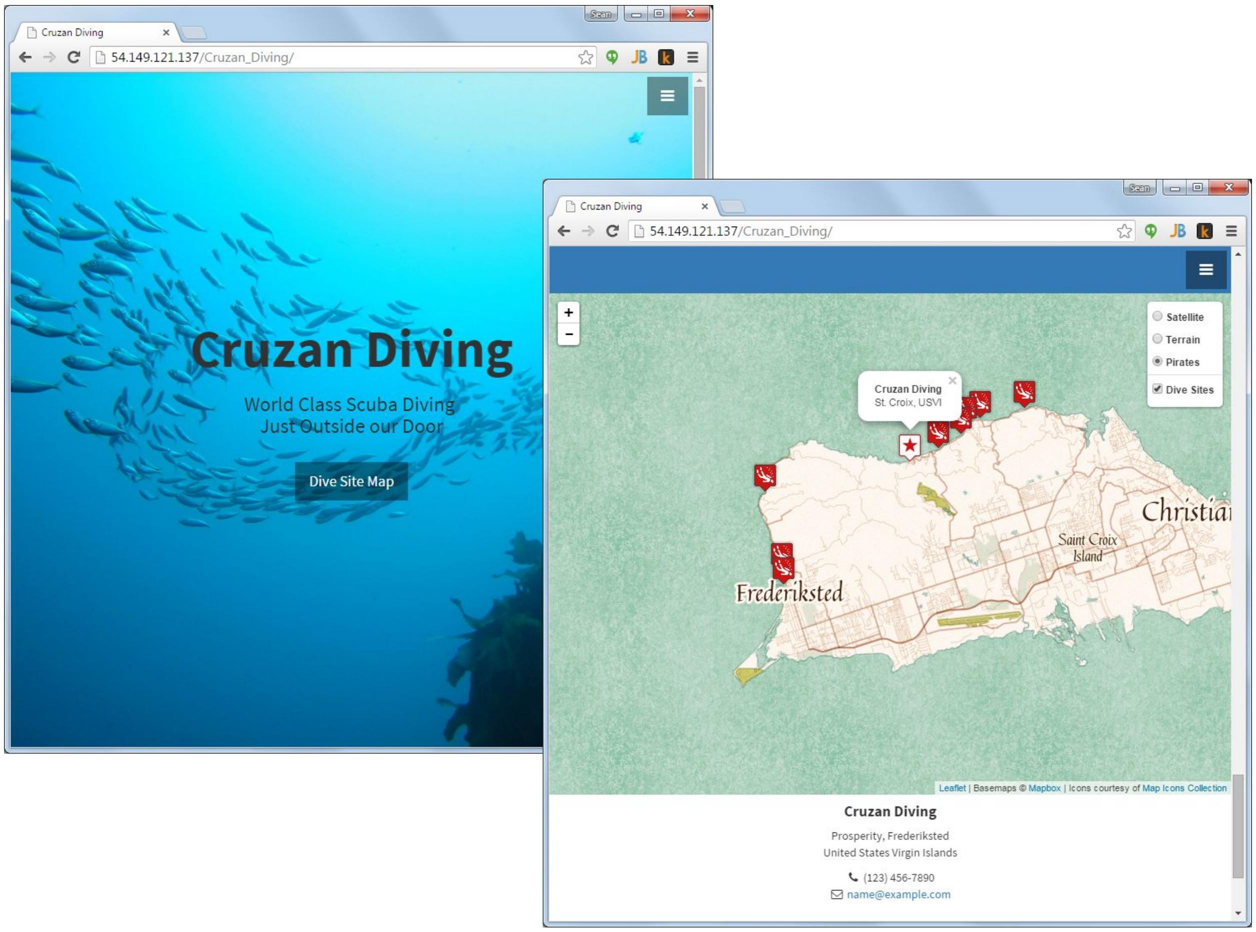

Cruzan Diving
localhost:63342/Cruzan_Diving/index.html
The buttons below are impossible to resist.
Click Me!
Look at Me!
Satellite
Terrain
Pirates
Dive Sites
Cruzan Diving
St. Croix, USVI
Frederiksted
Saint Croix Island
Christian
Leaflet | ACC GIS | Icons courtesy of Map Icons Collection | Map data © OpenStreetMap contributors, CC-BY-SA, Imagery © Mapbox
Cruzan Diving
Prosperity, Frederiksted
United States Virgin Islands
(123) 456-7890
name@example.com

Lifestyle business website with completed interactive map and toggle layer control

5. Publish Website with Embedded Map

Your lifestyle business website should now include a successfully embedded Leaflet interactive map that displays your location, nearby attractions, and Mapbox cached map services along with a toggle layer control.

Before publishing your map, you should breakout your CSS and JavaScript code to existing or new CSS and JavaScript files that are called from your HTML file. Confirm that your website and interactive map are still working and then publish your website using an Amazon Web Service (AWS) Elastic Cloud Computing (EC2) virtual server with Microsoft IIS. Submit your final website http address via Blackboard.



Business lifestyle responsive design website with interactive map of business location, nearby attractions, basemaps, and toggle layer control published on a cloud web server