Chapter 2 First Leaflet Map

2.1 Setting the superstructure

We had earlier mentioned that Javascript is the life of the party when it comes to webpages. In other words, it moves your web pages from a static sate to being interactive. You can think of it like the energy drink that makes an exhausted athlete want to do one more run.

Creating a Leaflet map is not like creating any other HTML web page. You have to set up the Leaflet essentials in your HTML page first. To begin with, create a new HTML document called `map.html` . This will be the HTML document that will act as the structure which will house our webpage to be created using JavaScript. Using VS Code, create `map.html` and paste, or preferably, type the following code.

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <title>Leaflet Maps</title>
        <meta charset="utf-8">
        <link rel="stylesheet" href="styles.css">
        <link rel="stylesheet" href="https://unpkg.com/leaflet@1.9.3/dist/leaflet.css"
     integrity="sha256-kLaT2GOSpHechhsozzB+flnD+zUyjE2LlfWPgU04xyI="
     crossorigin=""/>
        <script src="https://unpkg.com/leaflet@1.9.3/dist/leaflet.js"
        integrity="sha256-WBkoXOwTeyKclOHuWtc+i2uENFpDZ9YPdf5Hf+D7ewM="
        crossorigin=""></script>
    </head>
    <body>
        <div id="myMap">
            <script src="main.js">
            </script>
        </div>

    </body>
</html>
```

You may be wondering why we have two `<link>` and `<script>` tags. When a browser reads a HTML script, it reads it from top to bottom. In our html script, the browser will apply the styles defined in `styles.css` to the HTML elements. To make matters clearer, the following script is what is contained in the `styles.css` :

```
#myMap {
    height: 600px;
}
```

The CSS values, those within the curly brackets, will be responsible for making our map canvas have a height of 600px on the browser.

Now to the two `<script>` tags. One refers to the online JavaScript library. The `src` attribute is in fact linking to a webpage as you can see from the protocol– `https://...` . The second, housed under the `<div>` tag, references our local JavaScript file which shall contain all the code to transform our HTML page to a webmap ninja –lines, polygons and other cool stuff.

2.2 Beautifying the house

Think of the HTML document as the superstructure, like a huge multistorey building just finished. Though the structure has the best architectural design, it just looks all grey with no life unless we call some exterior designers to add some color. That's what `main.js` file, pointed to by the `<script>` tag in the HTML document will precisely do. Through the help of JavaScript, our static HTML file will turn into an interactive map.

Open your VS Code, and assuming you had already created `main.js` already, (if not, create one now), insert the following code into the `.js` file.

```
var map = L.map('myMap').setView([-0.0884105,34.7299038], 13);
```

Take a pause.

Breath in, breath out.

You are just about to learn something very important here. In fact, it is the crux of what makes Leaflet work. Your understanding of leaflet hinges on the small code above.

The `L.map()` class we just used is what initializes the Leaflet map. Everything within the `<div>` is displayed thanks to this class function. It is referred to as a factory function because it uses the method `map` to return an object.

The `setView` method *sets the view of the map (geographical center and zoom) with the given animation options*. It's properties are Latitude-Longitude, zoom number and other options. All Leaflet methods and functions are explained in the [Leaflet reference. Not that we have inserted at Latitude, Longitude and zoom level respectively within the `setView` method.

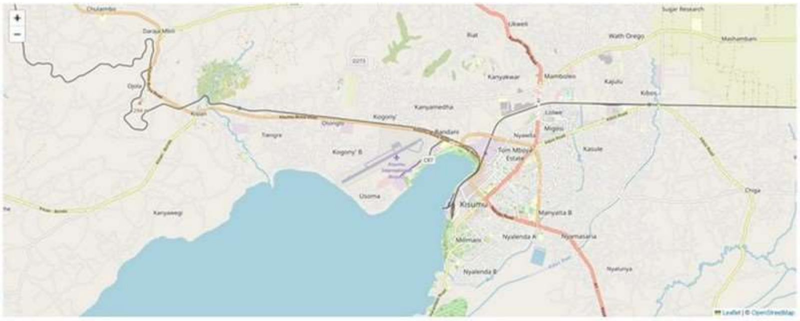
Try loading your `map.html` . All you see is a grey canvas with zoom options. This is because we haven't added a tilelayer yet. A tileLayer is a set of web-accessible tiles that reside on a server. A tile consists individual images or vector files from a server which are collectively joined together to form a webmap. If you've zoomed into a webmap, say Google Maps and noticed boxes appearing as you zoomed in or out, those are *tiles*.

Let's load an example of a common tile layer–the Open StreetMap–into Leaflet.

```
L.tileLayer('https://tile.openstreetmap.org/{z}/{x}/{y}.png', {
    maxZoom: 19,
    attribution: '&copy; <a href="http://www.openstreetmap.org/copyright">OpenStreetMap</a
}).addTo(map);
```

Reload your HTML page again. You should see an web map like the one below.

```
knitr::include_graphics(rep("D:/gachuhi/my-leaflet/images/kisumu-leaflet.jpg"))
```



What `L.tileLayer` has done is retrieve the web tiles from the Uniform Resource Locator (url) source provided, and within the dictionary that follows the url, zoom level (`maxZoom`) and map attribution (`attribution`) have been provided. When working with Leaflet, the dictionary, indicated by the curly braces `{}` houses most of the additional class options other than the key one(s). In this case we used the additional options of `maxZoom` and `attribution` . Finally, the method `addTo` adds the layer to the given map or layer group. Here, our webtile is added to the `var map` which only contains the `setView` properties.

A very influential person said Kisumu located in Kenya is a town with great potential. How about dispalying it to the whole world to realise it!

2.3 Summary

In this chapter, we created our first Leaflet map. Here are a couple of things that you have learnt at the first step of this web mapping journey.

- Browsers read code scripts from top to bottom, much like skimming down a page.
- The styles defined in the CSS style sheet, the `styles.css` in this case, will apply to the HTML elements in the `map.html` file.
- We use the `src` attribute to link a different file other than HTML, such as a JavaScript file, to your HTML file. For example, the online script `leaflet.js` is connected to the HTML file through the `src` attribute of the `<script>` to enable the HTML file execute Leaflet functionalities such as web map creation, rendering and controls.
- If HTML is the magnificent building, JavaScript acts like a good exterior designer.
- The `L.map()` class is what initializes the Leaflet map.