# SpringBoot Integration with Google Maps API

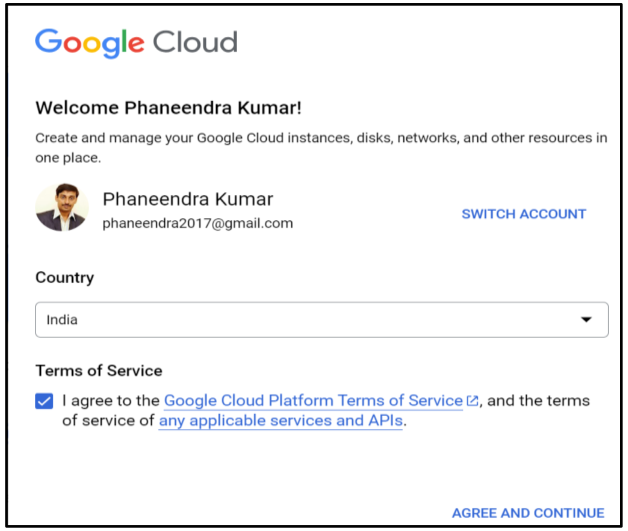
Dr. B L N Phaneendra Kumar (8215)
Assistant Professor
Department of Computer Science and Engineering
KL Deemed to be University
Vaddeswaram

# Steps to be done

- Generation of Google Map API Key
- Creation of Controller Java class
- Creation of index.html file which contain the information to be displayed on map

# Generation of Google Map API key

1. Goto website [https://console.cloud.google.com/getting-started](https://console.cloud.google.com/getting-started)
2. Login through your mail
3. Check the Terms of Service and click on Agree and continue
4. From the dropdown list 'Select a project' option



Google Cloud

**Welcome Phaneendra Kumar!**

Create and manage your Google Cloud instances, disks, networks, and other resources in one place.

Phaneendra Kumar
phaneendra2017@gmail.com

SWITCH ACCOUNT

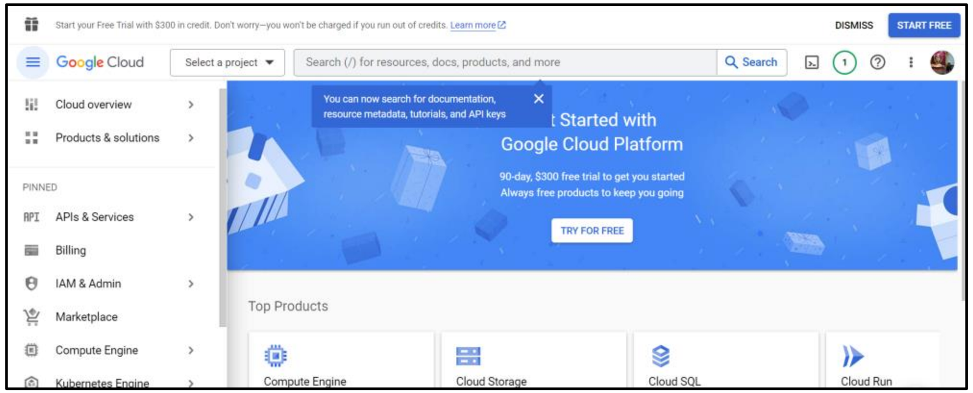
Country

India

Terms of Service

☑ I agree to the Google Cloud Platform Terms of Service, and the terms of service of any applicable services and APIs.

AGREE AND CONTINUE



Start your Free Trial with $300 in credit. Don't worry—you won't be charged if you run out of credits. Learn more

You can now search for documentation, resource metadata, tutorials, and API keys

t Started with Google Cloud Platform

90-day, $300 free trial to get you started
Always free products to keep you going

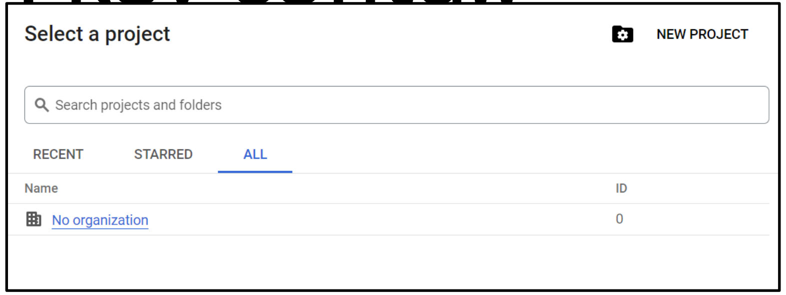
# Generation of Google Map API key contd..

5. Then, click on New Project (shown on top rightside)

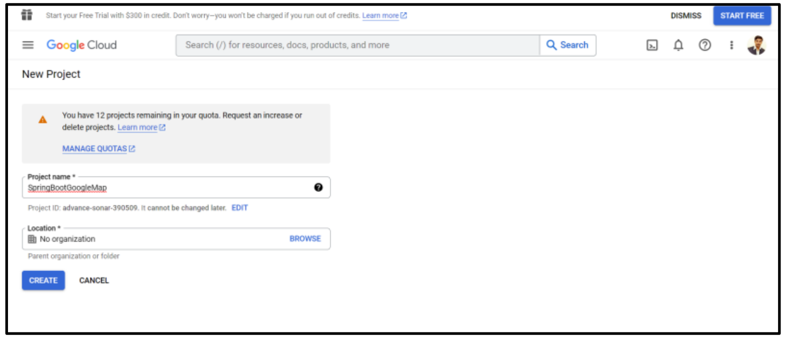
6. Provide a project name and click on Create button

7. On the navigation menu, Select 'APIs & Services' and from the list of options, select the Enabled APIs&services

8. Click on Enable on ENABLE APIS AND SERVICES



| Name | ID |
|---|---|
| No organization | 0 |



Start your Free Trial with $300 in credit. Don't worry—you won't be charged if you run out of credits. Learn more

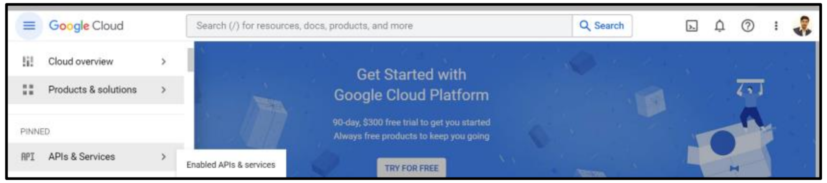
You have 12 projects remaining in your quota. Request an increase or delete projects. Learn more

Project name *: SpringBootGoogleMap

Project ID: advance-sonar-390509. It cannot be changed later. EDIT

Location *: No organization

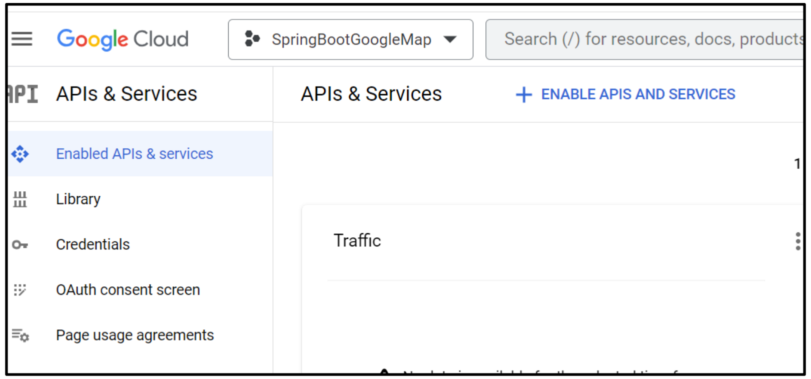
Parent organization or folder



Get Started with Google Cloud Platform

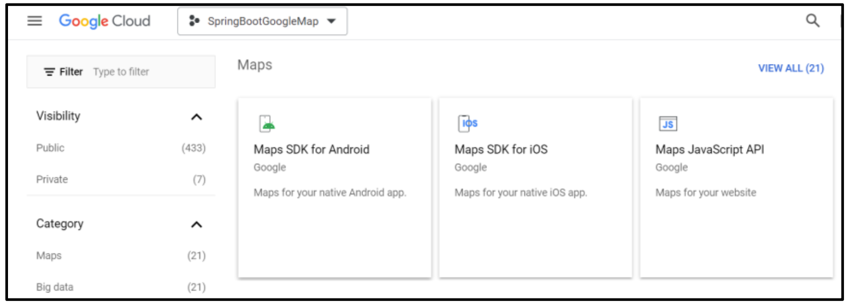
90-day, $300 free trial to get you started

Always free products to keep you going



# Generation of Google Map API key contd..

9. Then, Select Maps JavaScript API from the MAPs list

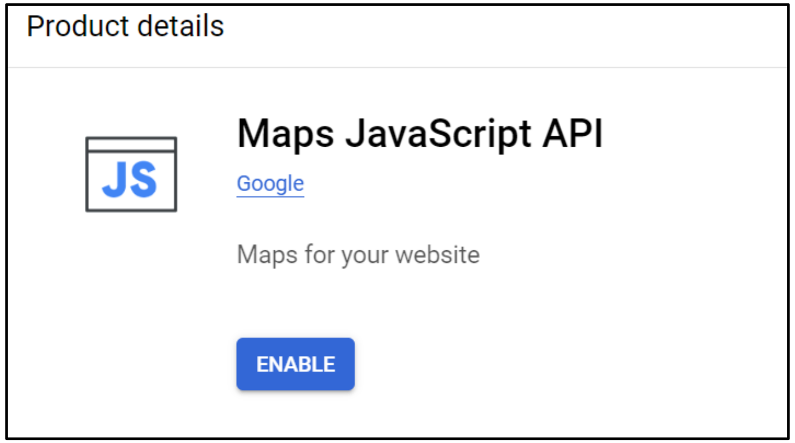


Maps SDK for Android
Google
Maps for your native Android app.

Maps SDK for iOS
Google
Maps for your native iOS app.

Maps JavaScript API
Google
Maps for your website

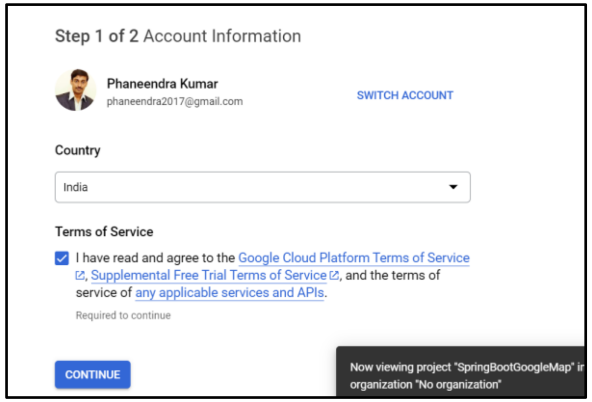
10. Click on ENABLE button



Maps JavaScript API

Maps for your website

11. Select the Check box Terms of Service and click on Continue



Step 1 of 2 Account Information

Phaneendra Kumar
phaneendra2017@gmail.com

Country
India

Terms of Service

☑ I have read and agree to the Google Cloud Platform Terms of Service, Supplemental Free Trial Terms of Service, and the terms of service of any applicable services and APIs.

Required to continue

Now viewing project "SpringBootGoogleMap" in organization "No organization"

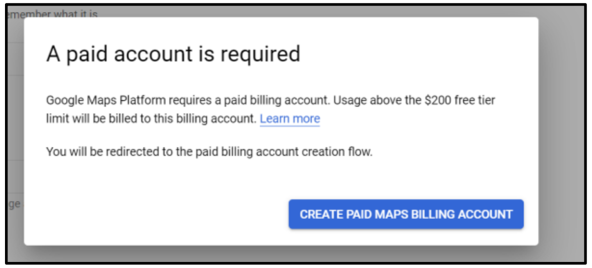
# Generation of Google Map API key contd..

12. It will ask for payment

13. Donot pay anything (Press Esc button)

14. Again click on Navigation Menu
    a. Select API&Services and go to Credentials

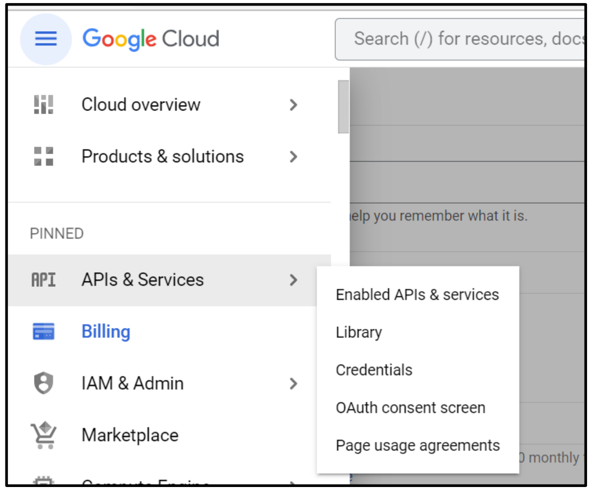
15. Click on Create Credentials

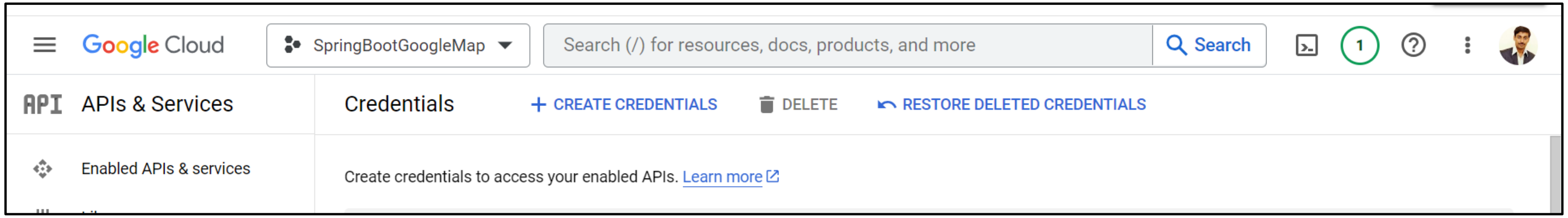


A paid account is required

Google Maps Platform requires a paid billing account. Usage above the $200 free tier limit will be billed to this billing account. Learn more

You will be redirected to the paid billing account creation flow.

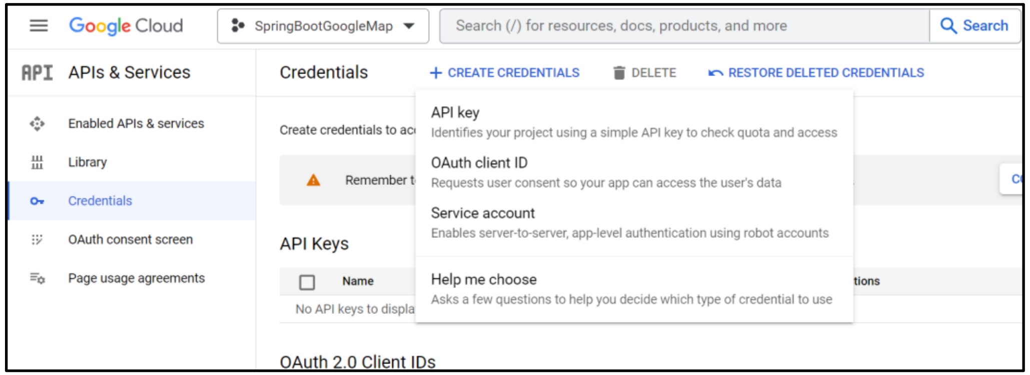




Create credentials to access your enabled APIs. Learn more

# Generation of Google Map API key contd..

- Then, Click on API Key



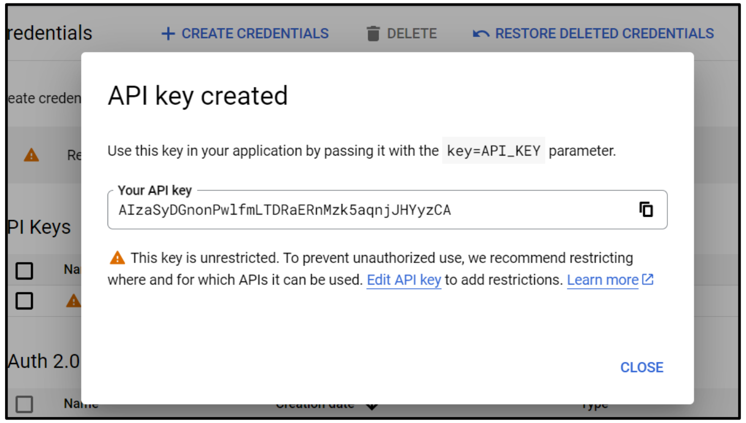
API key
Identifies your project using a simple API key to check quota and access

OAuth client ID
Requests user consent so your app can access the user's data

Service account
Enables server-to-server, app-level authentication using robot accounts

Help me choose
Asks a few questions to help you decide which type of credential to use

- It will create an API key
  - Copy and save it in a note pad



API key created

Use this key in your application by passing it with the `key=API_KEY` parameter.

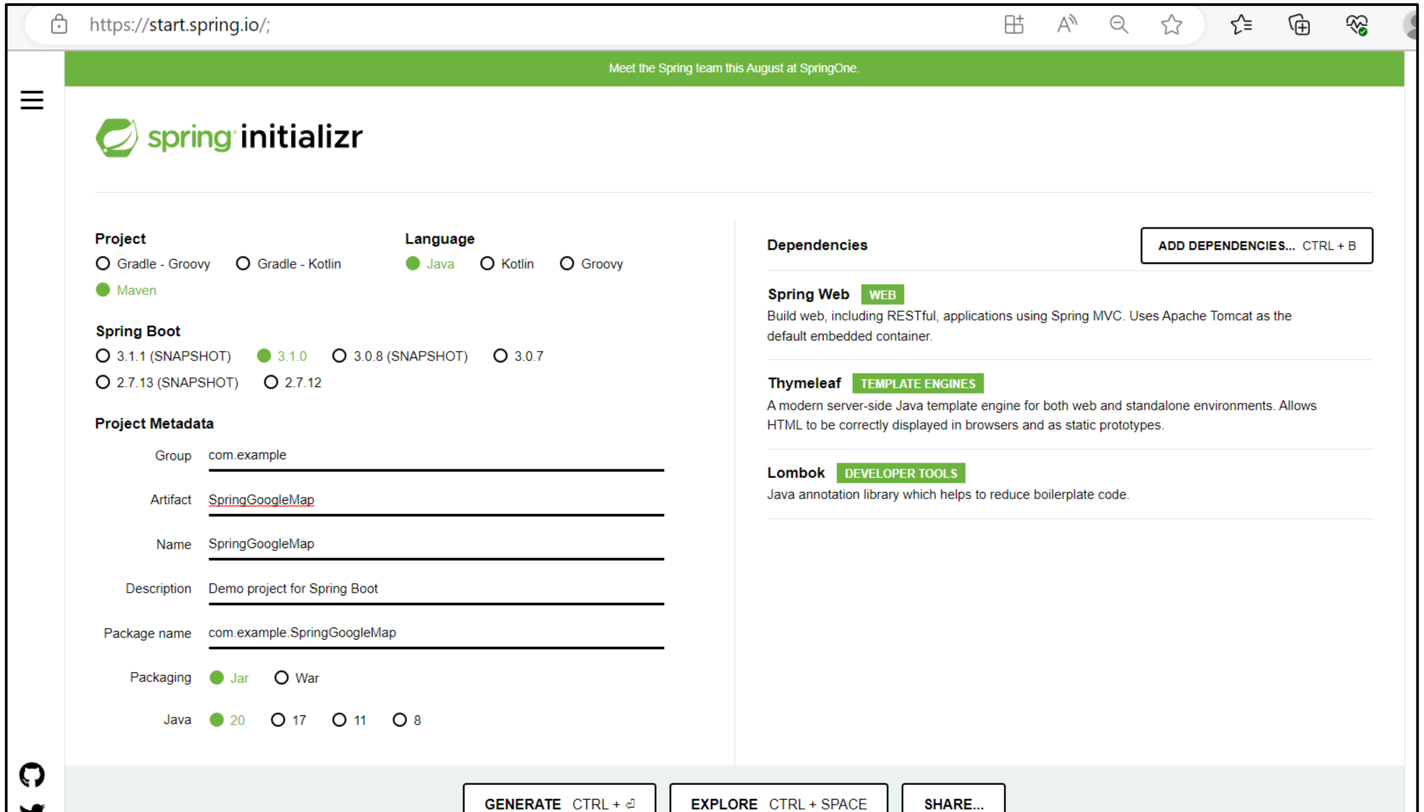
Your API key
`AIzaSyDGnonPwlfmLTDRaERnMzk5aqnjJHYyzCA`

⚠ This key is unrestricted. To prevent unauthorized use, we recommend restricting where and for which APIs it can be used. Edit API key to add restrictions. Learn more

- Then click on close

# Creation of SpringBoot Project

- Go to the SpringInitializr web site and create a project with necessary language, build tool, Java version and finally the required dependencies as shown below
- Then, Click on Generate



https://start.spring.io/;

Meet the Spring team this August at SpringOne.

spring initializr

**Project**
○ Gradle - Groovy ○ Gradle - Kotlin
● Maven

**Language**
● Java ○ Kotlin ○ Groovy

**Spring Boot**
○ 3.1.1 (SNAPSHOT) ● 3.1.0 ○ 3.0.8 (SNAPSHOT) ○ 3.0.7
○ 2.7.13 (SNAPSHOT) ○ 2.7.12

**Project Metadata**

Group: com.example

Artifact: SpringGoogleMap

Name: SpringGoogleMap

Description: Demo project for Spring Boot

Package name: com.example.SpringGoogleMap

Packaging: ● Jar ○ War

Java: ● 20 ○ 17 ○ 11 ○ 8

**Dependencies**

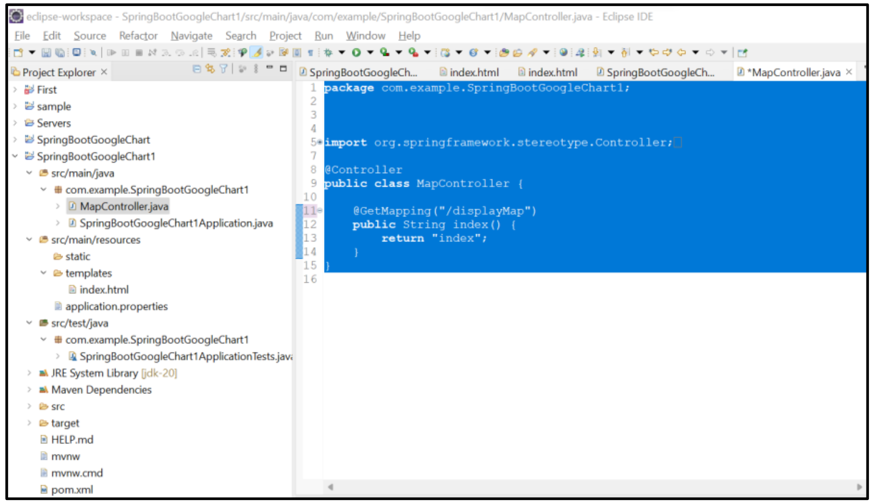
**Spring Web** WEB
Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.

**Thymeleaf** TEMPLATE ENGINES
A modern server-side Java template engine for both web and standalone environments. Allows HTML to be correctly displayed in browsers and as static prototypes.

**Lombok** DEVELOPER TOOLS
Java annotation library which helps to reduce boilerplate code.

# Files required to display Google Maps

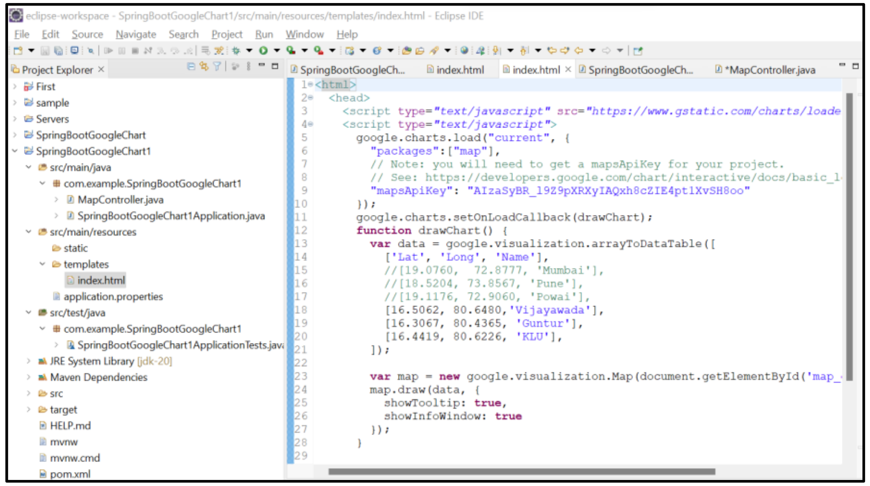
- Controller Class (MapController.java)
  - Create the java file under
  - Helps to invoke with the specified extension (@GetMapping)


eclipse-workspace - SpringBootGoogleChart1/src/main/java/com/example/SpringBootGoogleChart1/MapController.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
First
sample
Servers
SpringBootGoogleChart
SpringBootGoogleChart1
src/main/java
com.example.SpringBootGoogleChart1
MapController.java
SpringBootGoogleChart1Application.java
src/main/resources
static
templates
index.html
application.properties
src/test/java
com.example.SpringBootGoogleChart1
SpringBootGoogleChart1ApplicationTests.java
JRE System Library [jdk-20]
Maven Dependencies
src
target
HELP.md
mvnw
mvnw.cmd
pom.xml
SpringBootGoogleCh...
index.html
index.html
SpringBootGoogleCh...
*MapController.java

```java
package com.example.SpringBootGoogleChart1;



import org.springframework.stereotype.Controller;

@Controller
public class MapController {

    @GetMapping("/displayMap")
    public String index() {
        return "index";
    }
}
```

- Html file (index.html)
  - Under Templates create index.html

- Required dependencies such as SpringWeb, Thymeleaf and Lombok are added in pom.xml


eclipse-workspace - SpringBootGoogleChart1/src/main/resources/templates/index.html - Eclipse IDE
File Edit Source Navigate Search Project Run Window Help
Project Explorer
First
sample
Servers
SpringBootGoogleChart
SpringBootGoogleChart1
src/main/java
com.example.SpringBootGoogleChart1
MapController.java
SpringBootGoogleChart1Application.java
src/main/resources
static
templates
index.html
application.properties
src/test/java
com.example.SpringBootGoogleChart1
SpringBootGoogleChart1ApplicationTests.java
JRE System Library [jdk-20]
Maven Dependencies
src
target
HELP.md
mvnw
mvnw.cmd
pom.xml
SpringBootGoogleCh...
index.html
index.html
SpringBootGoogleCh...
*MapController.java

```html
<html>
  <head>
    <script type="text/javascript" src="https://www.gstatic.com/charts/loade
    <script type="text/javascript">
      google.charts.load("current", {
        "packages":["map"],
        // Note: you will need to get a mapsApiKey for your project.
        // See: https://developers.google.com/chart/interactive/docs/basic_l
        "mapsApiKey": "AIzaSyBR_19Z9pXRXyIAQxh8cZIE4pt1XvSH8oo"
      });
      google.charts.setOnLoadCallback(drawChart);
      function drawChart() {
        var data = google.visualization.arrayToDataTable([
          ['Lat', 'Long', 'Name'],
          //[19.0760,  72.8777, 'Mumbai'],
          //[18.5204, 73.8567, 'Pune'],
          //[19.1176, 72.9060, 'Powai'],
          [16.5062, 80.6480,'Vijayawada'],
          [16.3067, 80.4365, 'Guntur'],
          [16.4419, 80.6226, 'KLU'],
        ]);

        var map = new google.visualization.Map(document.getElementById('map_
        map.draw(data, {
          showTooltip: true,
          showInfoWindow: true
        });
      }

```

# MapController.java

```java
package com.example.SpringBootGoogleChart1;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.GetMapping;

@Controller
public class MapController {
@GetMapping("/displayMap")
    public String index() {
        return "index";
    }
}
```

# index.html (copy the copied googlemap api code in the html file)

```
<html>
  <head>
    <script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
    <script type="text/javascript">
      google.charts.load("current", {
        "packages":["map"],
        // Note: you will need to get a mapsApiKey for your project.
        // See: https://developers.google.com/chart/interactive/docs/basic_load_libs#load-settings
        "mapsApiKey": "AIzaSyDGnonPwlfmLTDRaERnMzk5aqnjJHYyzCA"
      });
      google.charts.setOnLoadCallback(drawChart);
      function drawChart() {
        var data = google.visualization.arrayToDataTable([
          ['Lat', 'Long', 'Name'],
          [16.5062, 80.6480,'Vijayawada'],
          [16.3067, 80.4365, 'Guntur'],
          [16.4419, 80.6226, 'KLU'],
        ]);
        var map = new google.visualization.Map(document.getElementById('map_div'));
        map.draw(data, {
          showTooltip: true,
          showInfoWindow: true
        });     }
    </script>
  </head>
  <body>
    <div id="map_div" style="width: 800px; height: 700px"></div>
  </body>
</html>
```

Assign the locations, we have to see on the google map

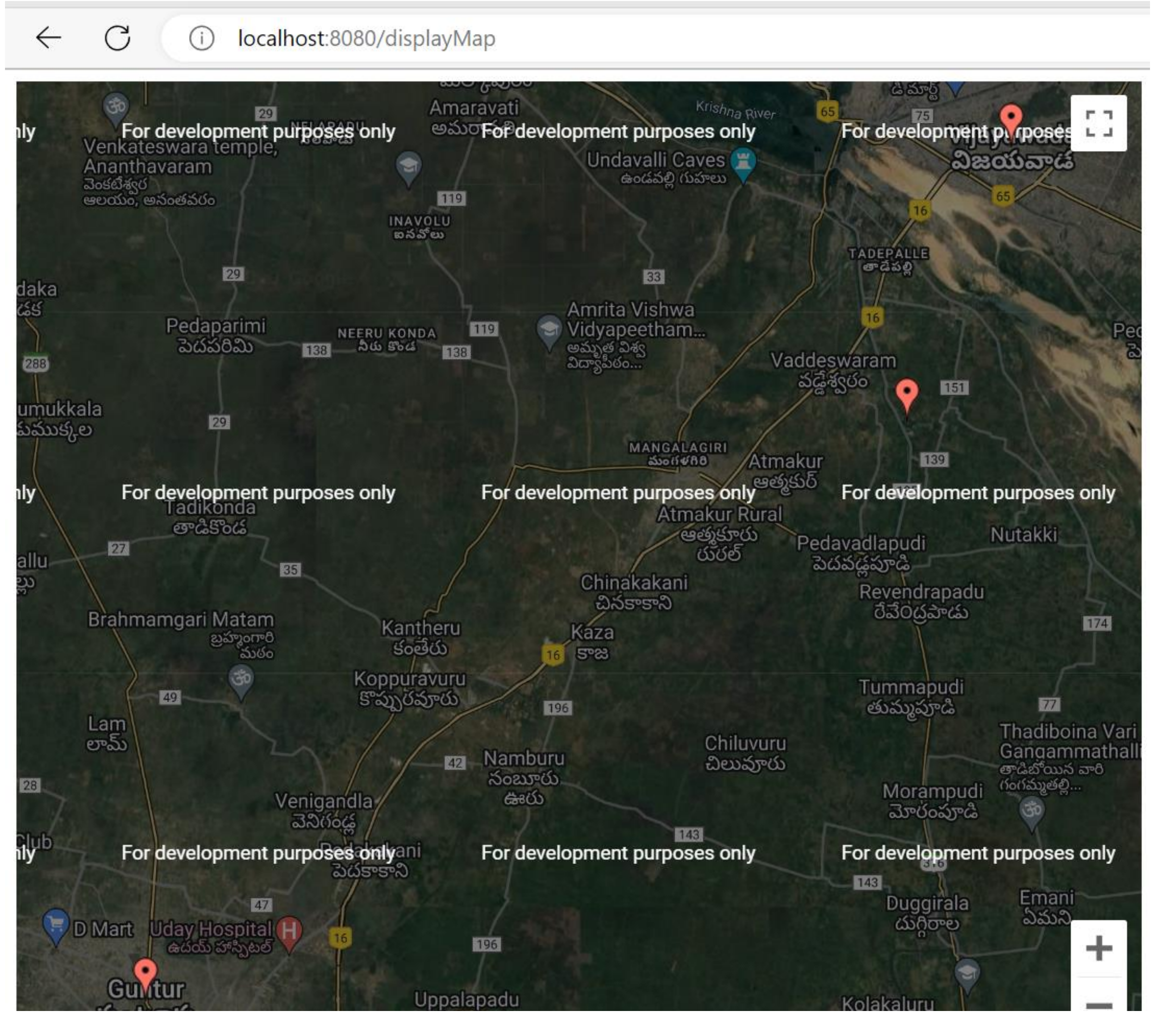
Enter longitute, latitude

# Execution

- Right click on the java file which contains the main method
    - **Run as Java Application**
- Then, Goto the browser and enter

  localhost:8080/displayMap

- We can see the output as shown here



localhost:8080/displayMap