Assignment

INTEGRATION OF API'S IN WEB PAGE

Name : Salman| Roll no: 2k17/IT/79 |
Class: BS-IT Part IV (Morning)
Semester: 8th | Assignment by:
Dr Kamran Taj Pathan

Assignment

Topic: Make a web services that is going to utilize any of the services of Profile (LinkedIn), Location (Google Maps), Time, Weather etc.

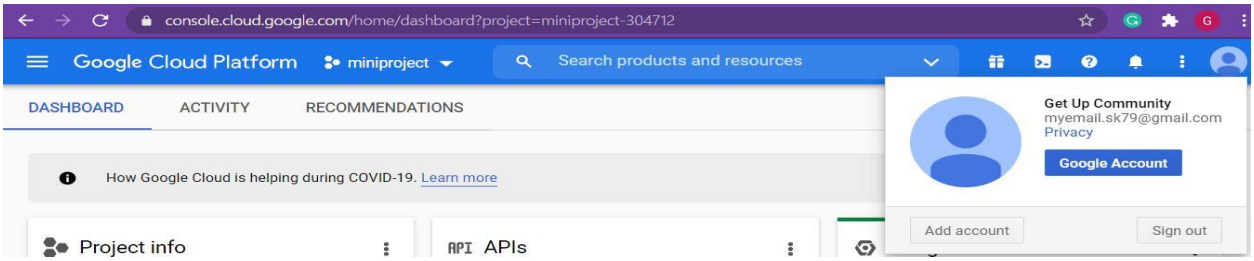
I Used 3 API's which is (Google maps API, Open Weather API, Google Sign-in API).

1. In this project I create a web page using html, JavaScript and css. Then integrate some different API's which collect current data and show on my web page.

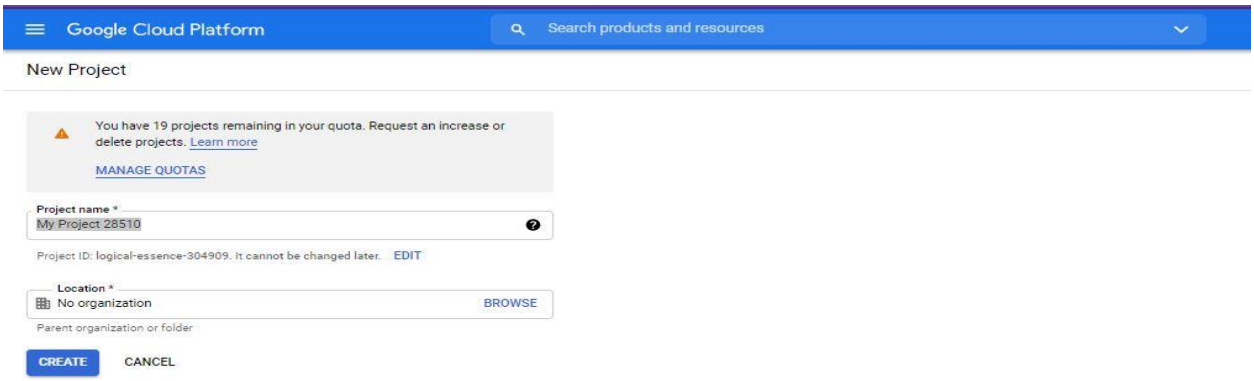
Firstly I used Google Maps API. I go to Google cloud platform and there I sign-in from my account. Below are some steps for integrating Google maps API on your website.

Steps:

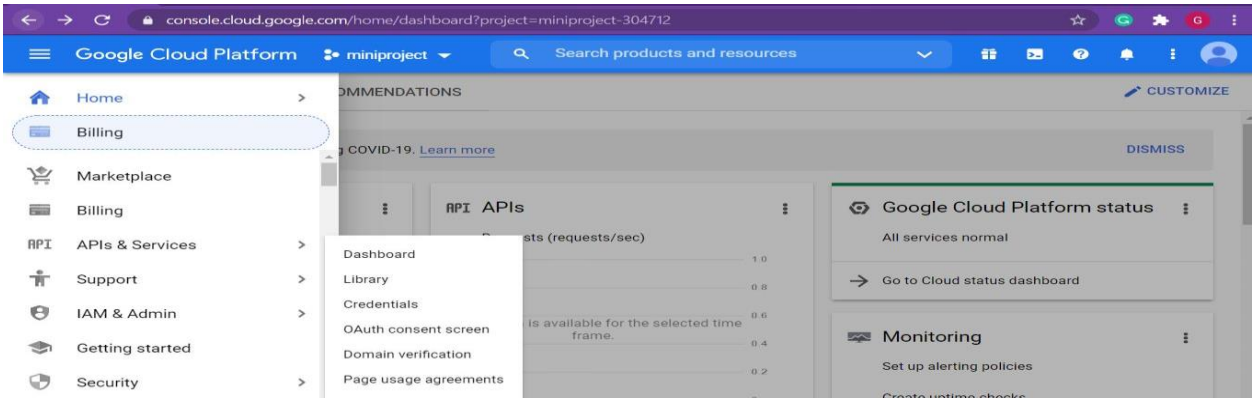
1. Sign-in in Google Cloud Platform

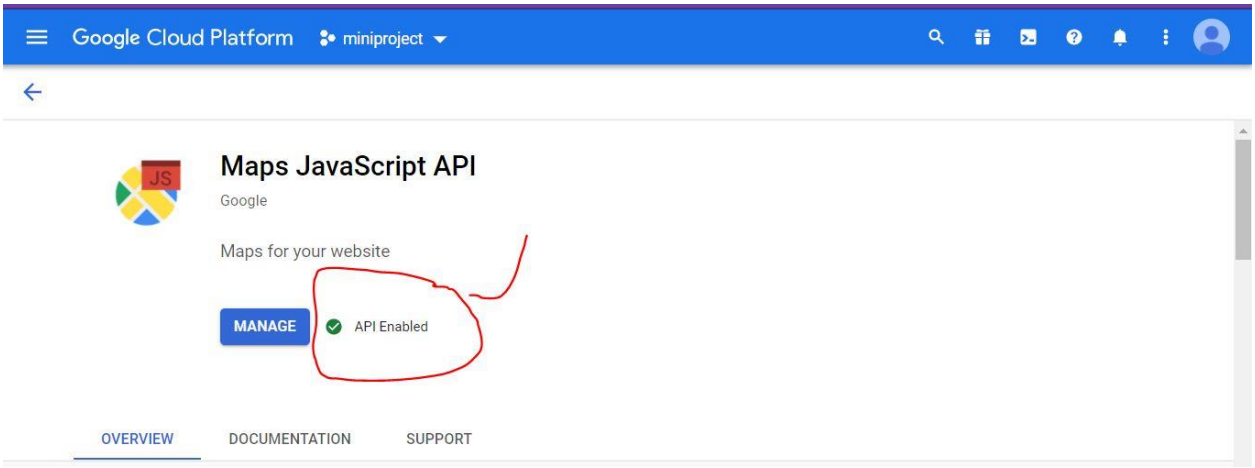


2. Create new project with suitable name.

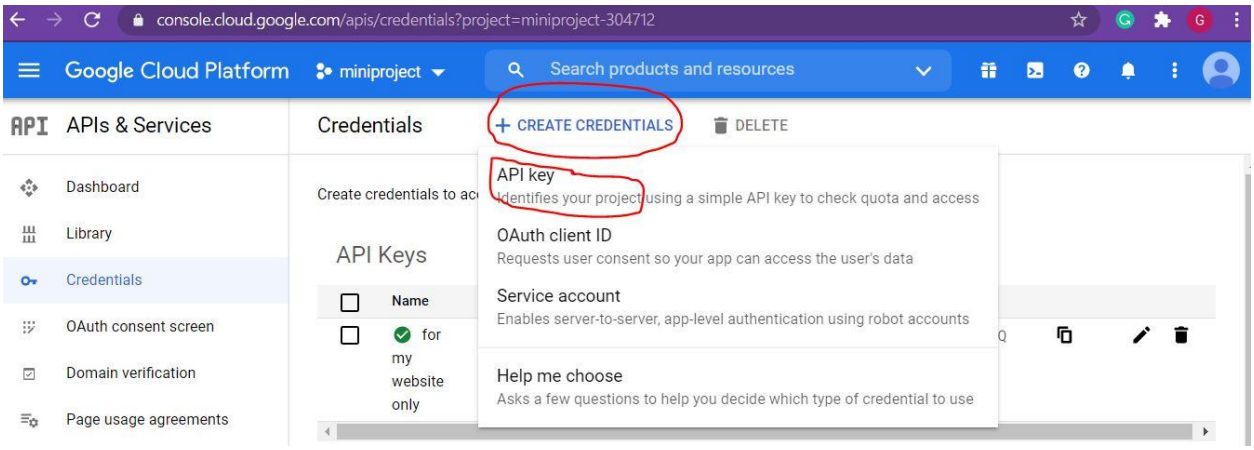


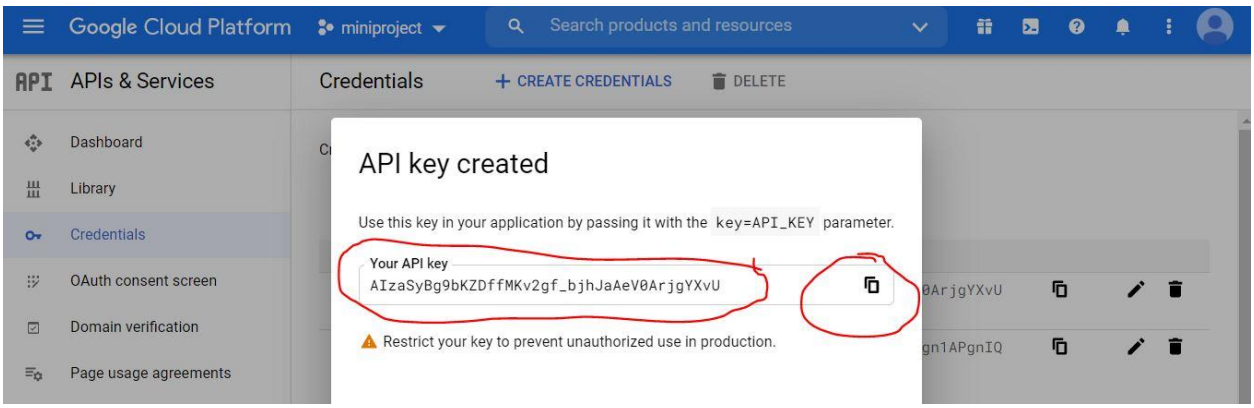
3. Go to menu then "API's and Service" then "Library" and search for (Google maps API JavaScript) and Enable (Google maps API JavaScript).





4. Then again go to menu click on <u>"API's and Service" then</u>" <u>Credentials"</u> create Credentials" Click on "API KEY" create API key And Copy This key And paste in the script Tag link of Google API.

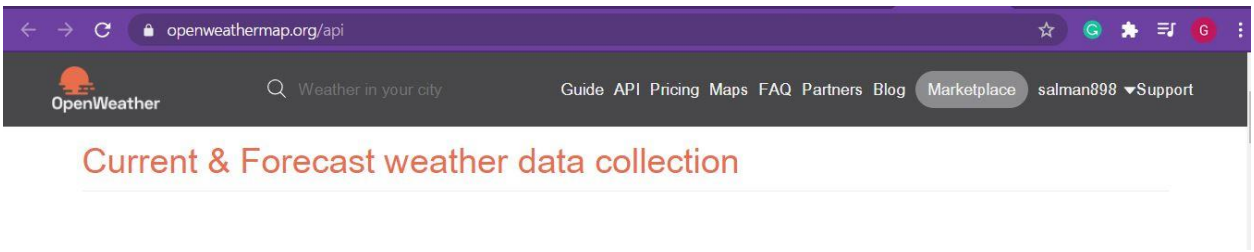




5. Put API key in script tag in the Google map API Link.

Used another weather API add:

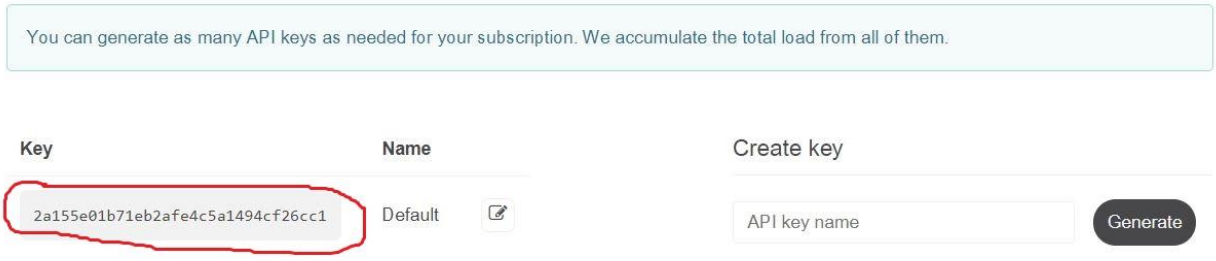
1. Go to (Openweathermap.org) site then sign-up here and then generate API Key And copy this key and put in your code javascript section.



Select Current weather API.

Generate and copy API KEY.

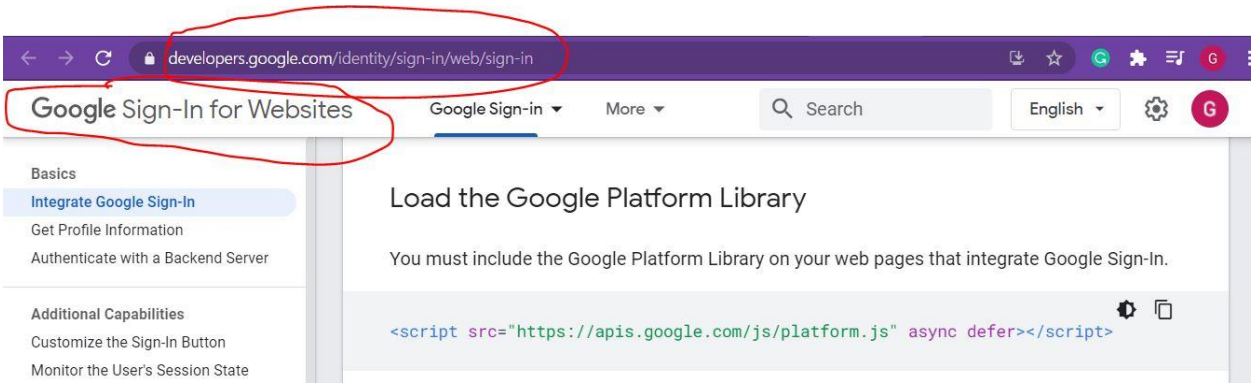
Weather API key Copy.



Paste API key in JavaScript Tag in the Link.

3rd API which I used in the page is "Google Sign-In":

Go to (https://developers.google.com/identity/sign-in/web/sign-in) and copy the Link for google Sign-in function.



Paste LINKs in your Script Tag section.

```
<!-- google sign-in script tag -->
<script src="https://apis.google.com/js/platform.js" async defer></script>
<meta name="google-signin-client_id" content="107362585837-tviaam6oc08r8gguidh99cj7r51685k5.apps.googleusercontent.com">
</head>
```

Project Code

index.html Page Code:

```
<!DOCTYPE html>
<html>
    <title>Weather app</title>
    <head>
        <link rel="stylesheet" href ="style.css">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <!-- google map scrip tag -->
        <script
      src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCvq33M0RmCwJiaVRotNMxAFI
OjGvEaUrI&callback=initMap&libraries=&v=weekly"
      defer
    ></script>
    <!-- google sign-in script tag -->
    <script src="https://apis.google.com/js/platform.js" async defer></script>
    <meta name="google-signin-client_id" content="107362585837-
tviaam6oc08r8gguidh99cj7r5l685k5.apps.googleusercontent.com">
    </head>

    <body>
        <!-- from here google maps api start -->
        <h1>GOOGLE MAP API</h1>
        <hr>
        <br/>
        <div id="map1"></div>


        <!-- from here google maps api End -->
        <h1>Weather API</h1>
        <hr>
        <form id="search-form">
            <input type="search"
            placeholder="Enter City Name"
            id="search-input"
            required
            autocomplete="off"
            />
            <br>
        </br>
        <button id="search-button">Search</button>

        </form>


        <main id="app-container">
            <div id="location">
                <p>-------</p>
            </div>
                <div id="temp">
```

```
                    <img id="temp-icon" src="./icons/sun.svg" alt="">
                    <p><span id="temp-value">-----</span> <span id="temp-
unit" >&#176c</span> </p>
                </div>

                <div id="climate">
                    <p>------</p>
                </div>

        </main>
        <!-- from here Google sign -->
        <h1>Sign-in Api</h1>
    <hr>
    <br/>
  <div class="g-signin2" data-onsuccess="onSignIn"></div>

<script src="script.js">

</script>
    </body>

</html>
```

script.js Page Code:

```
// from here maps javacript start
let map;
function initMap() {
  map = new google.maps.Map(document.getElementById("map1"), {
    center: { lat: 25.3960, lng: 68.3578 },
    zoom: 8,
    mapTypeId: "terrain"
  });
  const marker = new google.maps.Marker({
      position: { lat: 25.3960, lng: 68.3578 },
      map: map,
      label: "A",
      title: " New Delhi",
      draggable: false,
      animation: google.maps.Animation.DROP,
      icon: "map.png"
  });

  const infoWindow = new google.maps.InfoWindow({
    content: "<p>This is an info window</p>"
  });
  infoWindow.open(map, marker);
}
// from here maps javacript End
```

```
// weather api javascript
let loc =document.getElementById("location");
let tempicon=document.getElementById("temp-icon");
let tempvalue=document.getElementById("temp-value");
let climate =document.getElementById("climate");
let iconfile;
const searchInput=document.getElementById("search-input");
const searchButton=document.getElementById("search-button");

searchButton.addEventListener('click', (e)=>
{

e.preventDefault();
getWeather(searchInput.value);
searchInput.value='';


});



const getWeather=async (city)=>
{
    try{

        const response= await fetch(`https://api.openweathermap.org/data/2.5/weat
her?q=${city}&appid=dab3af44de7d24ae7ff86549334e45bd`,

            {mode: 'cors'}
        );

        const weatherData= await response.json();
        console.log(weatherData);
        const{name}=weatherData;
        const{feels_like}=weatherData.main;
        const{id,main}=weatherData.weather[0];
        loc.textContent=name;
        climate.textContent=main;
        tempvalue.textContent=Math.round(feels_like-273);
        if(id<300 && id>200)
        {
            tempicon.src="./icons/thunderstorm.svg"
        }
       else  if(id<400 && id>300)
        {
            tempicon.src="./icons/cloud-solid.svg"
        }
       else if(id<600&& id>500)
        {
            tempicon.src="./icons/rain.svg"
        }
```

```
        else  if(id<700 && id>600)
         {
             tempicon.src="./icons/snow.svg"
         }
        else  if(id<800 && id>700)
         {
             tempicon.src="./icons/clouds.svg"
         }
          else if(id==800)
         {
             tempicon.src="./icons/clouds-and-sun.svg"
         }
    }
catch(error)
{
    alert('city not found');
}
};

window.addEventListener("load" ,()=>{

let long;
let lat;

if(navigator.geolocation)
{

    navigator.geolocation.getCurrentPosition((position)=>
    {
    long=position.coords.longitude;
    lat=position.coords.latitude;
    const proxy="https://cors-anywhere.herokuapp.com/";

        const api=`${proxy}api.openweathermap.org/data/2.5/weather?lat=${lat}&lon
=${long}&appid=dab3af44de7d24ae7ff86549334e45bd      `

        fetch(api).then((response)=>{

            return response.json();

        })

        .then (data =>
            {

                    const{name}=data;
                    const{feels_like}=data.main;
                    const{id,main}=data.weather[0];
                    loc.textContent=name;
                    climate.textContent=main;
```

```
                    tempvalue.textContent=Math.round(feels_like-273);
                    if(id<300 && id>200)
                    {
                        tempicon.src="./icons/thunderstorm.svg"
                    }
                   else  if(id<400 && id>300)
                    {
                        tempicon.src="./icons/cloud-solid.svg"
                    }
                   else if(id<600&& id>500)
                    {
                        tempicon.src="./icons/rain.svg"
                    }
                   else  if(id<700 && id>600)
                    {
                        tempicon.src="./icons/snow.svg"
                    }
                   else  if(id<800 && id>700)
                    {
                        tempicon.src="./icons/clouds.svg"
                    }
                     else if(id==800)
                    {
                        tempicon.src="./icons/clouds-and-sun.svg"
                    }
                    console.log(data);
            })
}
    )}
})
```

style.css Page Code:

```
*{
        box-sizing: border-box;
        margin: 0;
        padding: 0;
}
body{
    height: 100px;
    background-color: #263859;
}

#map1 {
    height: 400%;
    width: 90%;
    margin: auto;
  }
h1 {
    text-align: center;
```

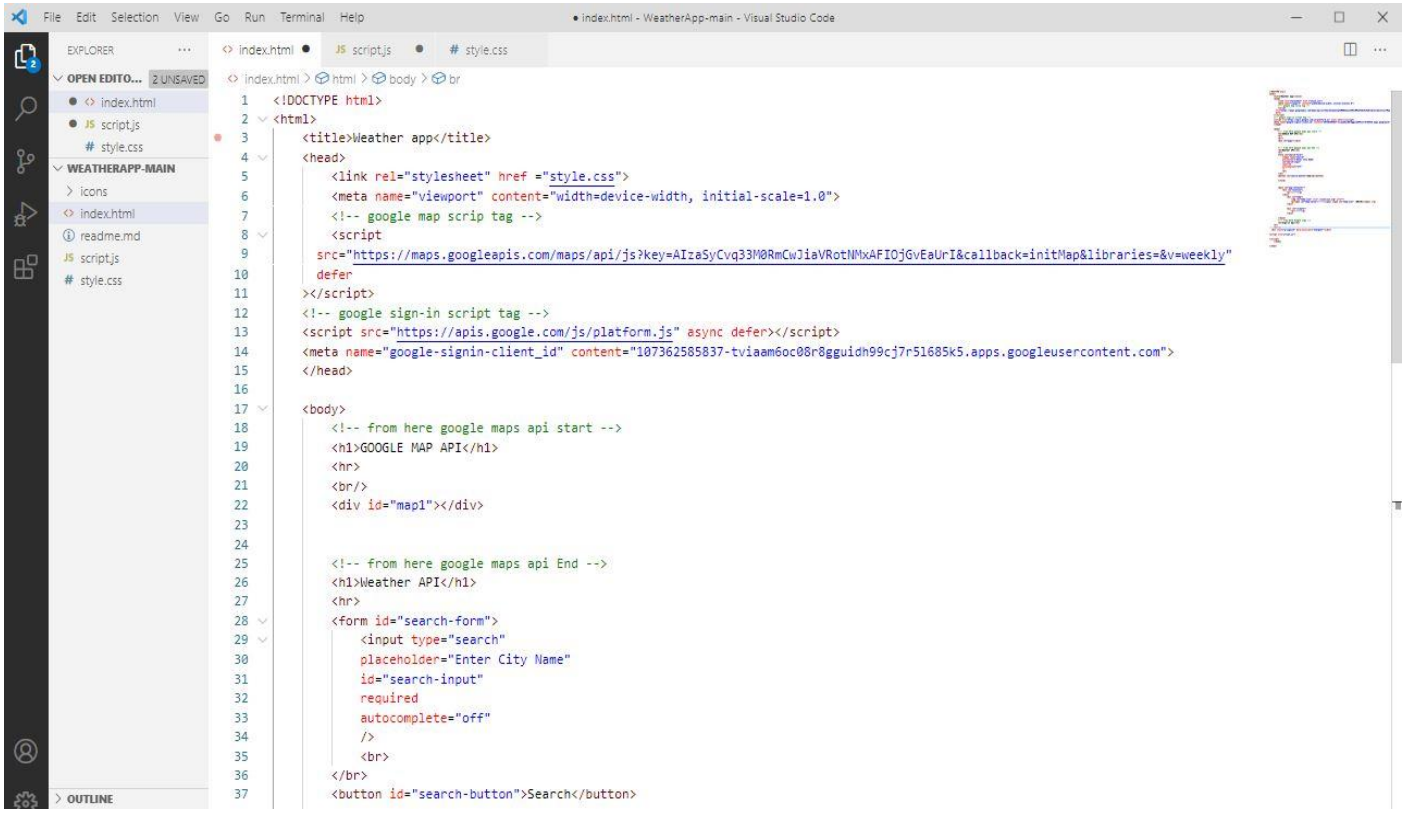
```
}
#search-form
{
    display: flex;
    justify-content: center;
    align-items: center;
    margin-top: 100px;
}

#temp
{
text-align: center;
}

#temp #temp-unit
{
    font-family: -apple-
system, BlinkMacSystemFont, 'Segoe UI', Roboto, Oxygen, Ubuntu, Cantarell, 'Open
Sans', 'Helvetica Neue', sans-serif;
    font-size: 1.5rem;
    color: #ffffff;
}
#temp #temp-value
{
    font-family: -apple-
system, BlinkMacSystemFont, 'Segoe UI', Roboto, Oxygen, Ubuntu, Cantarell, 'Open
Sans', 'Helvetica Neue', sans-serif;
    font-size: 2rem;
    color: #ffffff;

}
#climate
{
    font-family: -apple-
system, BlinkMacSystemFont, 'Segoe UI', Roboto, Oxygen, Ubuntu, Cantarell, 'Open
Sans', 'Helvetica Neue', sans-serif;
    font-size: 2.5rem;
    color: #ffffff;
    text-align: center;
}
#location
{
    font-family: -apple-
system, BlinkMacSystemFont, 'Segoe UI', Roboto, Oxygen, Ubuntu, Cantarell, 'Open
Sans', 'Helvetica Neue', sans-serif;
    font-size: 2.5rem;
    color: #ffffff;
    text-align: center;
}
```

```css
#temp-icon
{
    margin: 1rem;
    width: 40%;
    height: auto;
}


main{

    padding: 20px;
    margin-top: 10px;
    background-color: #6b778d;
    border-radius: 20px;
    width: 360px;
    height: 360px;
    display: block;
    margin-left: auto;
    margin-right: auto;
    box-shadow: 10px 10px 20px 10px rgba(112, 102, 102, 0.2);

}


#search-button
{
    color: #494949;
    text-transform: uppercase;
    text-decoration: none;
    background: white;
    border-radius: 5px;
    margin-left: 5px;
    padding: 11px;
    border: 2px solid #494949;
    display: inline-block;
    transition: all 0.4 ease 0s;
}
#search-button:hover{

color: #ffffff;
background: #ff6768;
border-color: #ff6768;
transition: all 0.4s ease 0s;



}
```

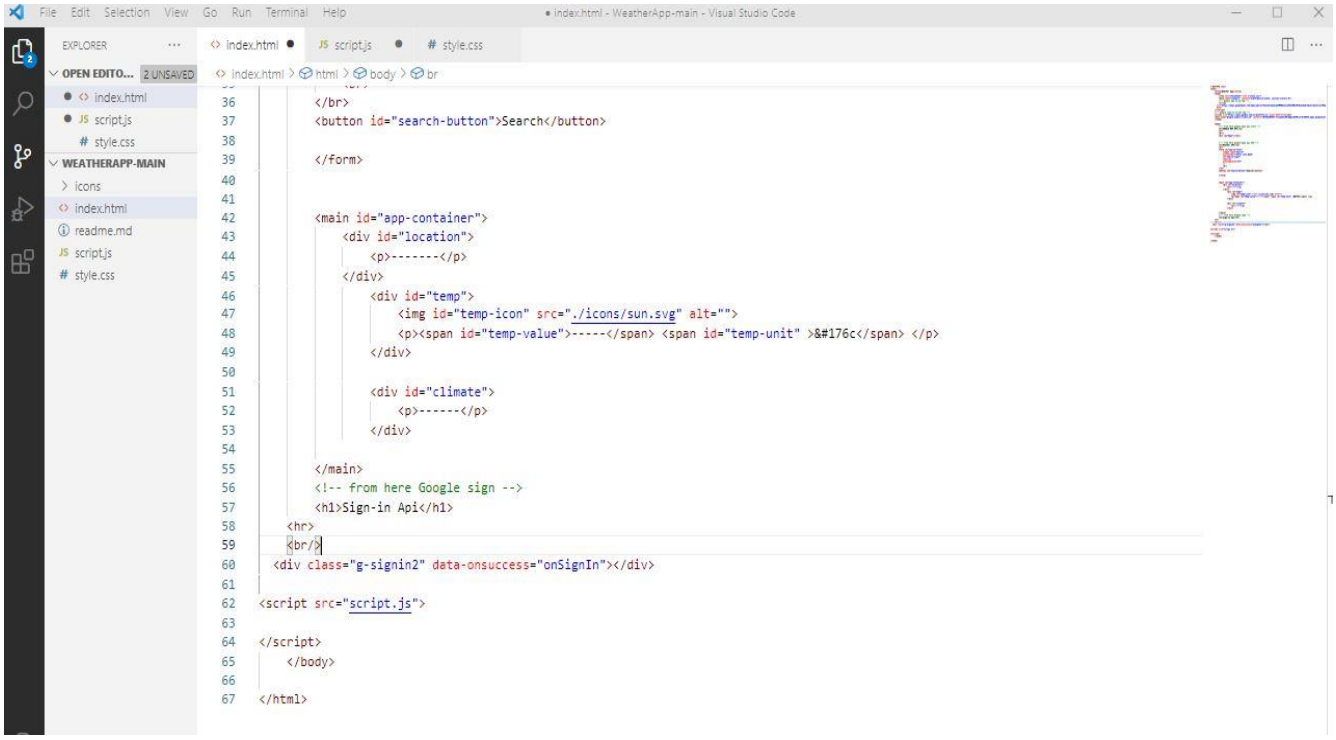
```css
#search-input
{
    width: 260px;
    overflow: hidden;
    font-size: 20px;
    margin: 8px 0;
    padding: 8px 0;
    border-bottom: 3px solid #ff6768;
    color: white;
    outline: none;
    background: none;
}
input
{
    border: none;
}
/* google sign-in css */

.g-signin2{
    margin-left: 50%;
}
/* .g-signin2 .onSignIn{
    height: 100%;
    width: 50%;
} */
```

API's Code ScreenShot's

index.html code Screenshot:

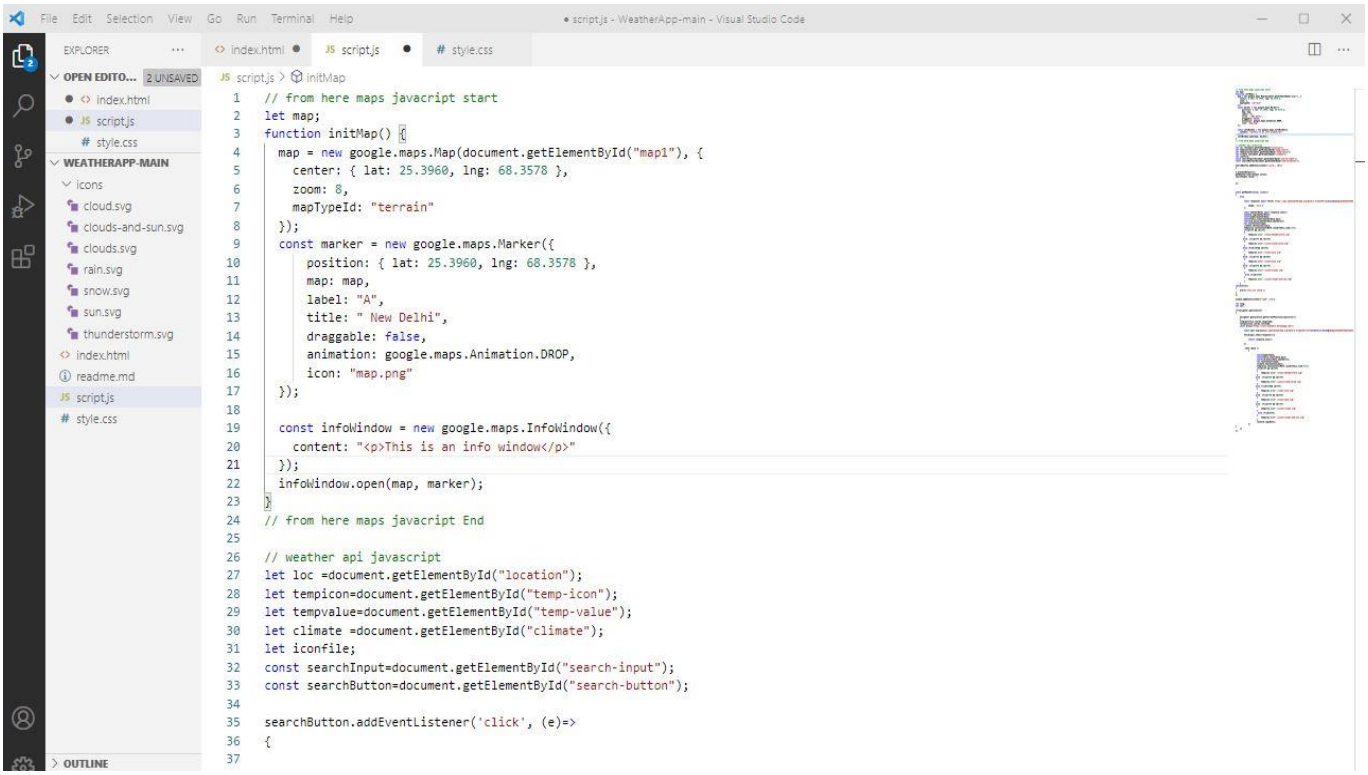


```
<!DOCTYPE html>
<html>
    <title>Weather app</title>
    <head>
        <link rel="stylesheet" href ="style.css">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <!-- google map scrip tag -->
        <script
      src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCvq33M0RmCwJiaVRotNMxAFIOjGvEaUrI&callback=initMap&libraries=&v=weekly"
      defer
    ></script>
    <!-- google sign-in script tag -->
    <script src="https://apis.google.com/js/platform.js" async defer></script>
    <meta name="google-signin-client_id" content="107362585837-tviaam6oc08r8gguidh99cj7r51685k5.apps.googleusercontent.com">
    </head>

    <body>
        <!-- from here google maps api start -->
        <h1>GOOGLE MAP API</h1>
        <hr>
        <br/>
        <div id="map1"></div>


        <!-- from here google maps api End -->
        <h1>Weather API</h1>
        <hr>
        <form id="search-form">
            <input type="search"
            placeholder="Enter City Name"
            id="search-input"
            required
            autocomplete="off"
            />
            <br>
        </br>
        <button id="search-button">Search</button>
```

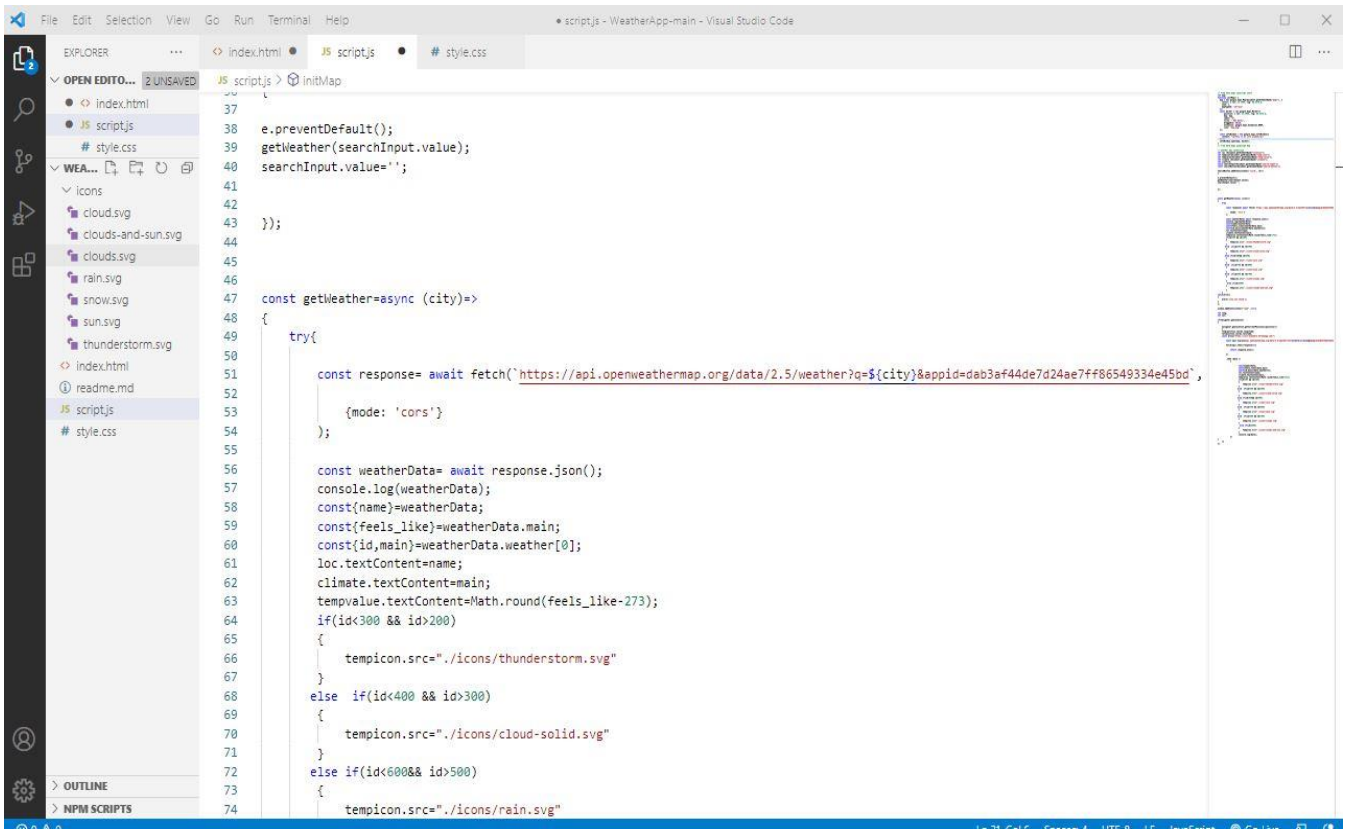


```
        </br>
        <button id="search-button">Search</button>

        </form>


        <main id="app-container">
            <div id="location">
                <p>-------</p>
            </div>
                <div id="temp">
                    <img id="temp-icon" src="./icons/sun.svg" alt="">
                    <p><span id="temp-value">-----</span> <span id="temp-unit" >&#176c</span> </p>
                </div>

                <div id="climate">
                    <p>------</p>
                </div>

        </main>
        <!-- from here Google sign -->
        <h1>Sign-in Api</h1>
    <hr>
    <br/>
  <div class="g-signin2" data-onsuccess="onSignIn"></div>

<script src="script.js">

</script>
    </body>

</html>
```

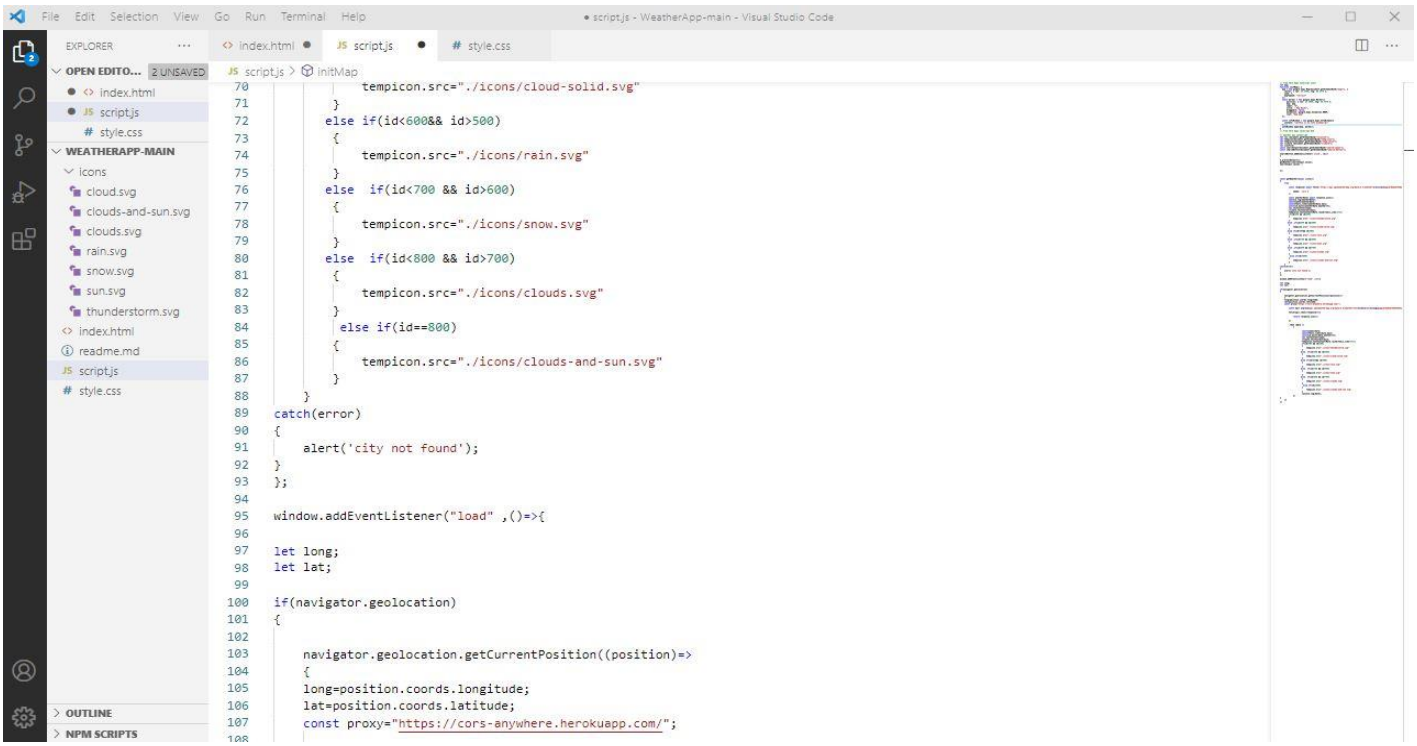
script.js Code ScreenShot's:



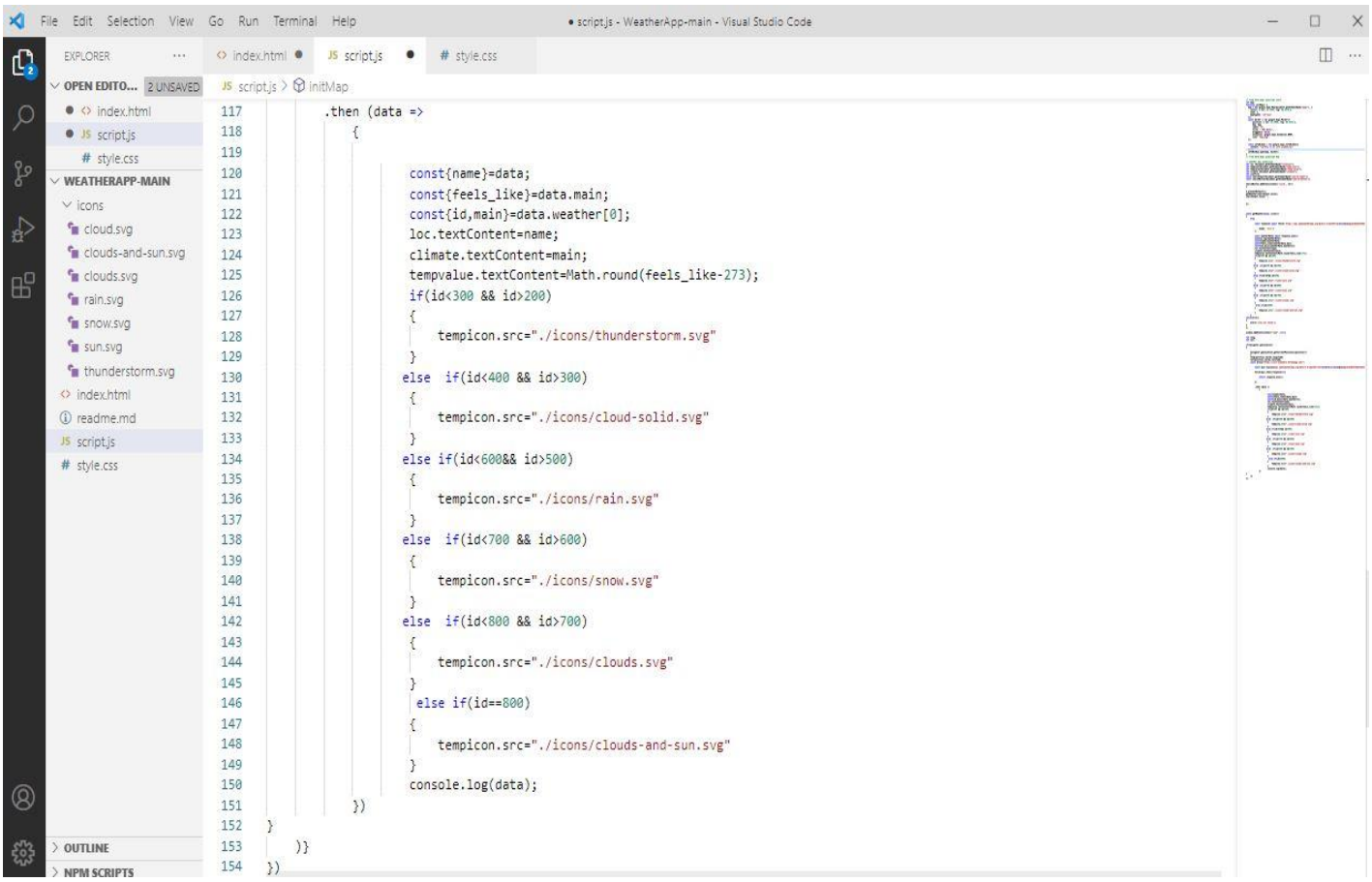
```
// from here maps javacript start
let map;
function initMap() {
  map = new google.maps.Map(document.getElementById("map1"), {
    center: { lat: 25.3960, lng: 68.3578 },
    zoom: 8,
    mapTypeId: "terrain"
  });
  const marker = new google.maps.Marker({
      position: { lat: 25.3960, lng: 68.3578 },
      map: map,
      label: "A",
      title: " New Delhi",
      draggable: false,
      animation: google.maps.Animation.DROP,
      icon: "map.png"
  });

  const infoWindow = new google.maps.InfoWindow({
    content: "<p>This is an info window</p>"
  });
  infoWindow.open(map, marker);
}
// from here maps javacript End

// weather api javascript
let loc =document.getElementById("location");
let tempicon=document.getElementById("temp-icon");
let tempvalue=document.getElementById("temp-value");
let climate =document.getElementById("climate");
let iconfile;
const searchInput=document.getElementById("search-input");
const searchButton=document.getElementById("search-button");

searchButton.addEventListener('click', (e)=>
{
```



```
e.preventDefault();
getWeather(searchInput.value);
searchInput.value='';


});



const getWeather=async (city)=>
{
    try{

        const response= await fetch(`https://api.openweathermap.org/data/2.5/weather?q=${city}&appid=dab3af44de7d24ae7ff86549334e45bd`,

            {mode: 'cors'}
        );

        const weatherData= await response.json();
        console.log(weatherData);
        const{name}=weatherData;
        const{feels_like}=weatherData.main;
        const{id,main}=weatherData.weather[0];
        loc.textContent=name;
        climate.textContent=main;
        tempvalue.textContent=Math.round(feels_like-273);
        if(id<300 && id>200)
        {
            tempicon.src="./icons/thunderstorm.svg"
        }
       else  if(id<400 && id>300)
        {
            tempicon.src="./icons/cloud-solid.svg"
        }
       else if(id<600&& id>500)
        {
            tempicon.src="./icons/rain.svg"
```

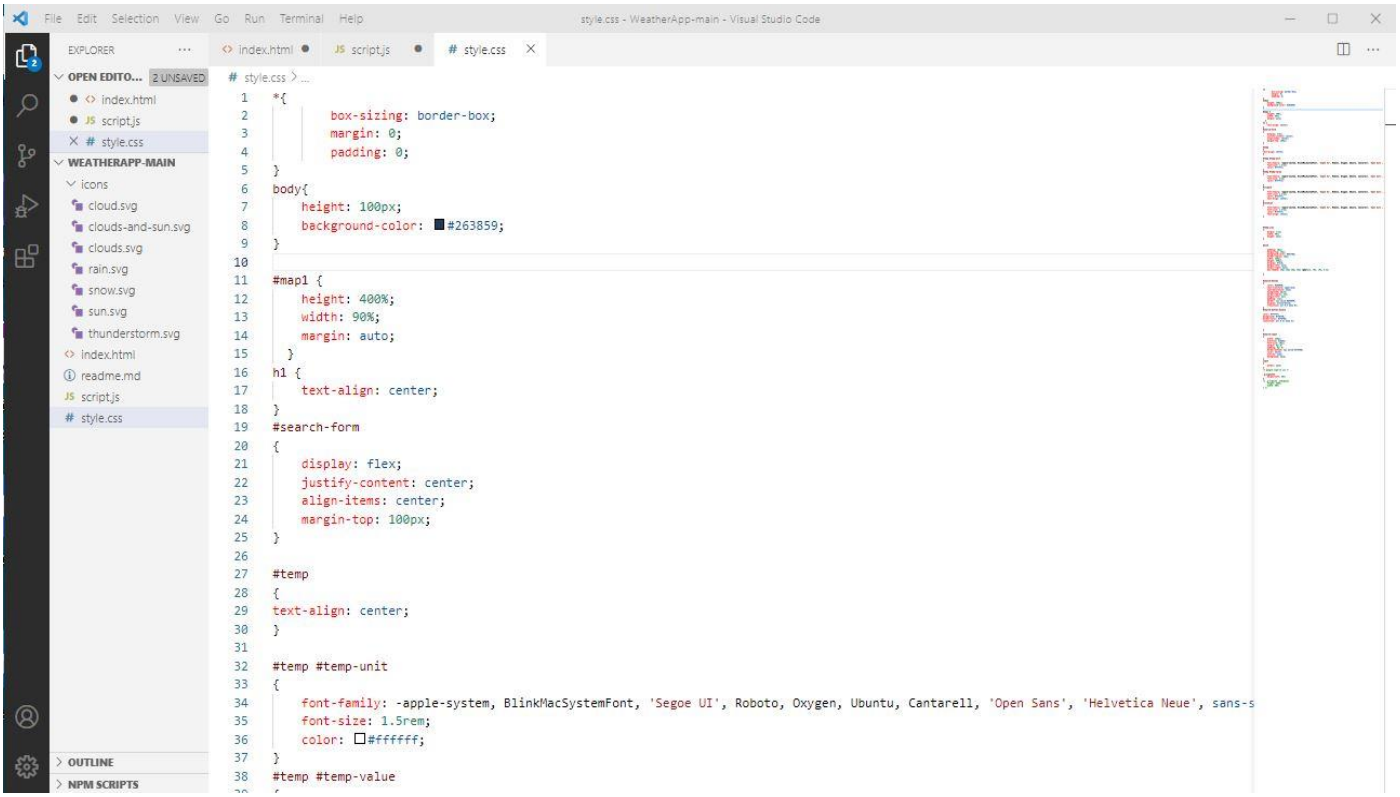


```
            tempicon.src="./icons/cloud-solid.svg"
        }
        else if(id<600&& id>500)
        {
            tempicon.src="./icons/rain.svg"
        }
        else  if(id<700 && id>600)
        {
            tempicon.src="./icons/snow.svg"
        }
        else  if(id<800 && id>700)
        {
            tempicon.src="./icons/clouds.svg"
        }
         else if(id==800)
        {
            tempicon.src="./icons/clouds-and-sun.svg"
        }
    }
catch(error)
{
    alert('city not found');
}
};

window.addEventListener("load" ,()=>{

let long;
let lat;

if(navigator.geolocation)
{

    navigator.geolocation.getCurrentPosition((position)=>
    {
    long=position.coords.longitude;
    lat=position.coords.latitude;
    const proxy="https://cors-anywhere.herokuapp.com/";
```

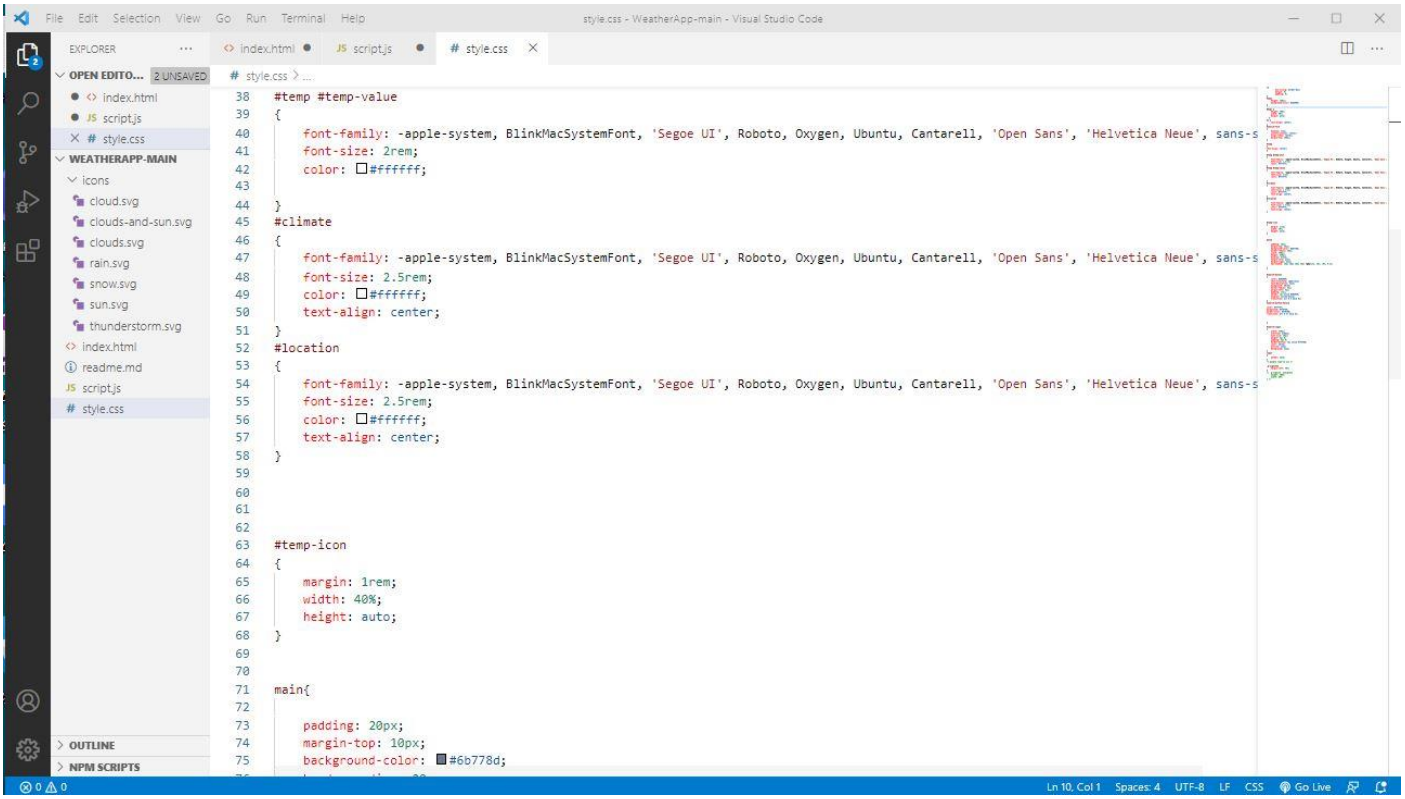


```
        .then (data =>
            {

                const{name}=data;
                const{feels_like}=data.main;
                const{id,main}=data.weather[0];
                loc.textContent=name;
                climate.textContent=main;
                tempvalue.textContent=Math.round(feels_like-273);
                if(id<300 && id>200)
                {
                    tempicon.src="./icons/thunderstorm.svg"
                }
                else  if(id<400 && id>300)
                {
                    tempicon.src="./icons/cloud-solid.svg"
                }
                else if(id<600&& id>500)
                {
                    tempicon.src="./icons/rain.svg"
                }
                else  if(id<700 && id>600)
                {
                    tempicon.src="./icons/snow.svg"
                }
                else  if(id<800 && id>700)
                {
                    tempicon.src="./icons/clouds.svg"
                }
                 else if(id==800)
                {
                    tempicon.src="./icons/clouds-and-sun.svg"
                }
                console.log(data);
            })
}
    )}
})
```

style.css Code ScreenShot's:



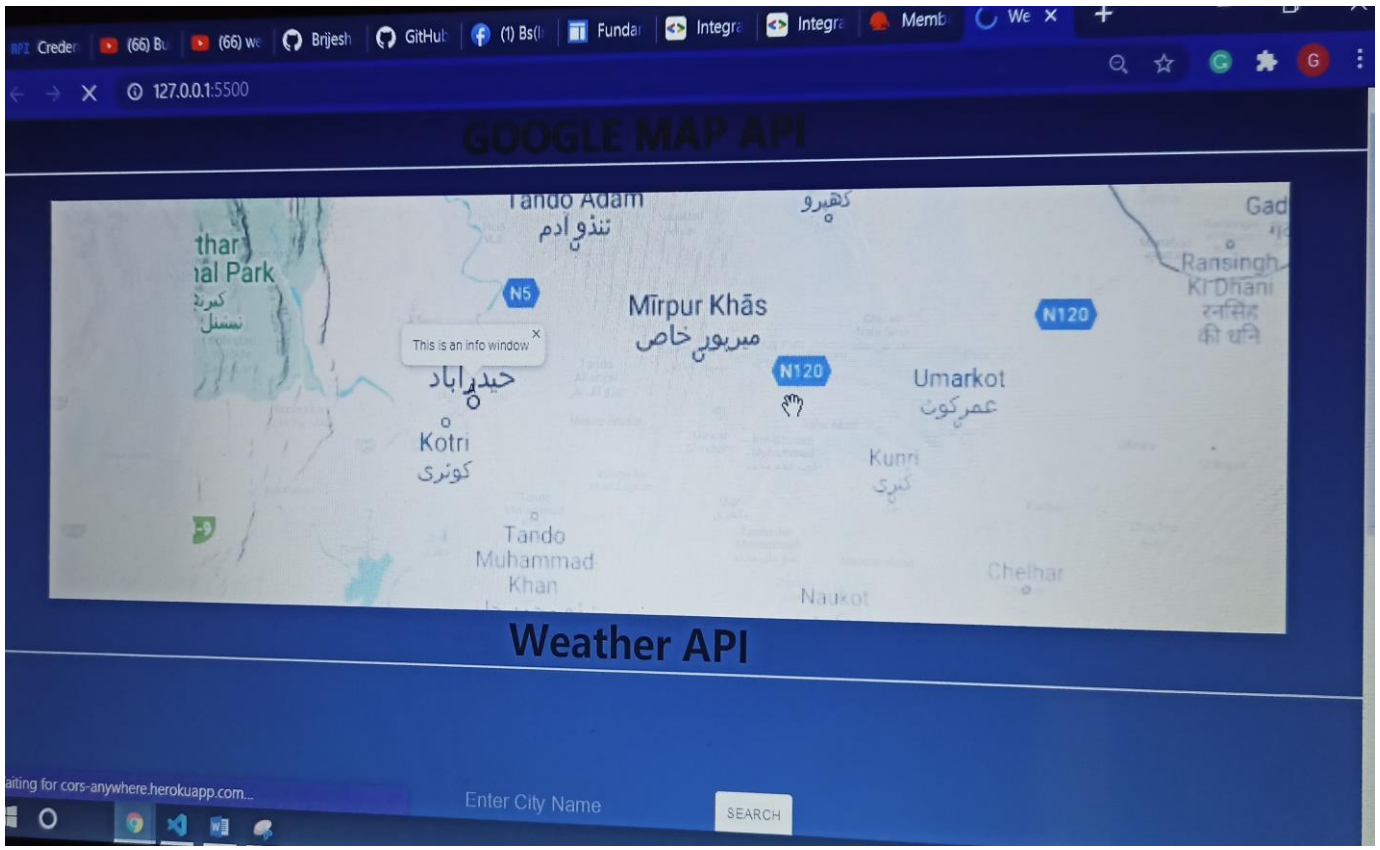
```
*{
        box-sizing: border-box;
        margin: 0;
        padding: 0;
}
body{
    height: 100px;
    background-color: #263859;
}

#map1 {
    height: 400%;
    width: 90%;
    margin: auto;
  }
h1 {
    text-align: center;
}
#search-form
{
    display: flex;
    justify-content: center;
    align-items: center;
    margin-top: 100px;
}

#temp
{
text-align: center;
}

#temp #temp-unit
{
    font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto, Oxygen, Ubuntu, Cantarell, 'Open Sans', 'Helvetica Neue', sans-s
    font-size: 1.5rem;
    color: #ffffff;
}
#temp #temp-value
```



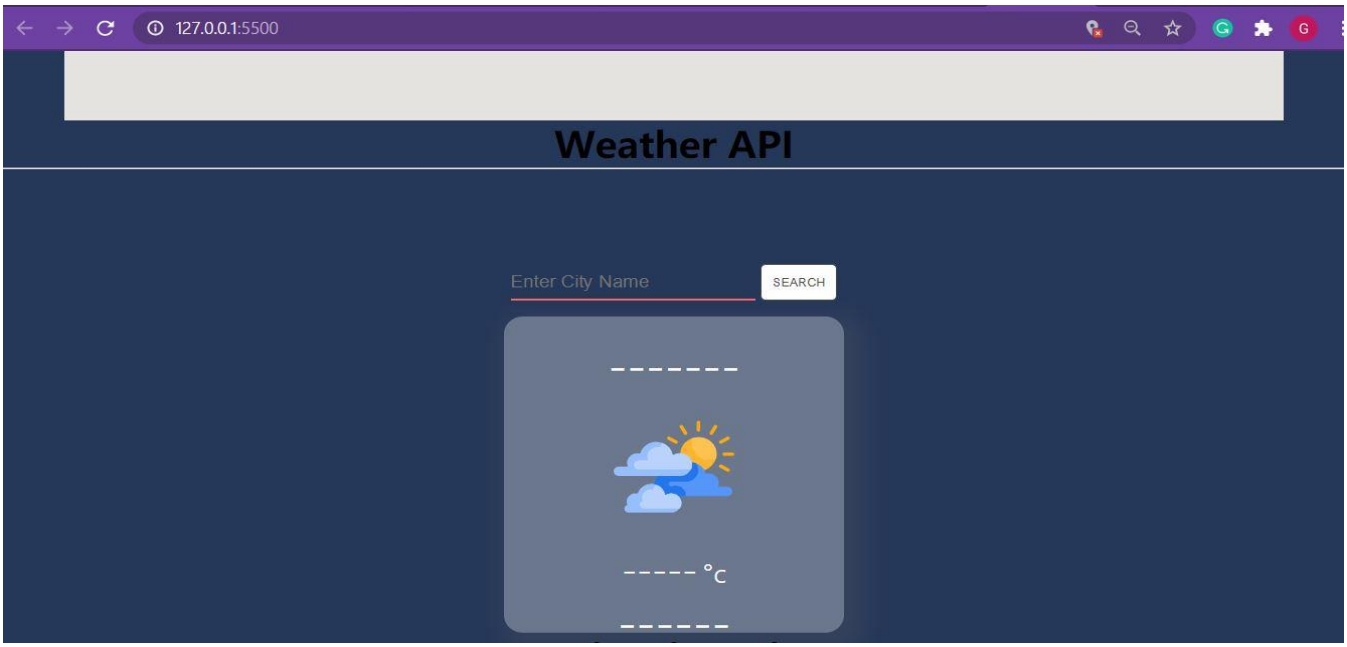
```
#temp #temp-value
{
    font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto, Oxygen, Ubuntu, Cantarell, 'Open Sans', 'Helvetica Neue', sans-s
    font-size: 2rem;
    color: #ffffff;

}
#climate
{
    font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto, Oxygen, Ubuntu, Cantarell, 'Open Sans', 'Helvetica Neue', sans-s
    font-size: 2.5rem;
    color: #ffffff;
    text-align: center;
}
#location
{
    font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto, Oxygen, Ubuntu, Cantarell, 'Open Sans', 'Helvetica Neue', sans-s
    font-size: 2.5rem;
    color: #ffffff;
    text-align: center;
}




#temp-icon
{
    margin: 1rem;
    width: 40%;
    height: auto;
}


main{

    padding: 20px;
    margin-top: 10px;
    background-color: #6b778d;
```

API Project Result or Outputs

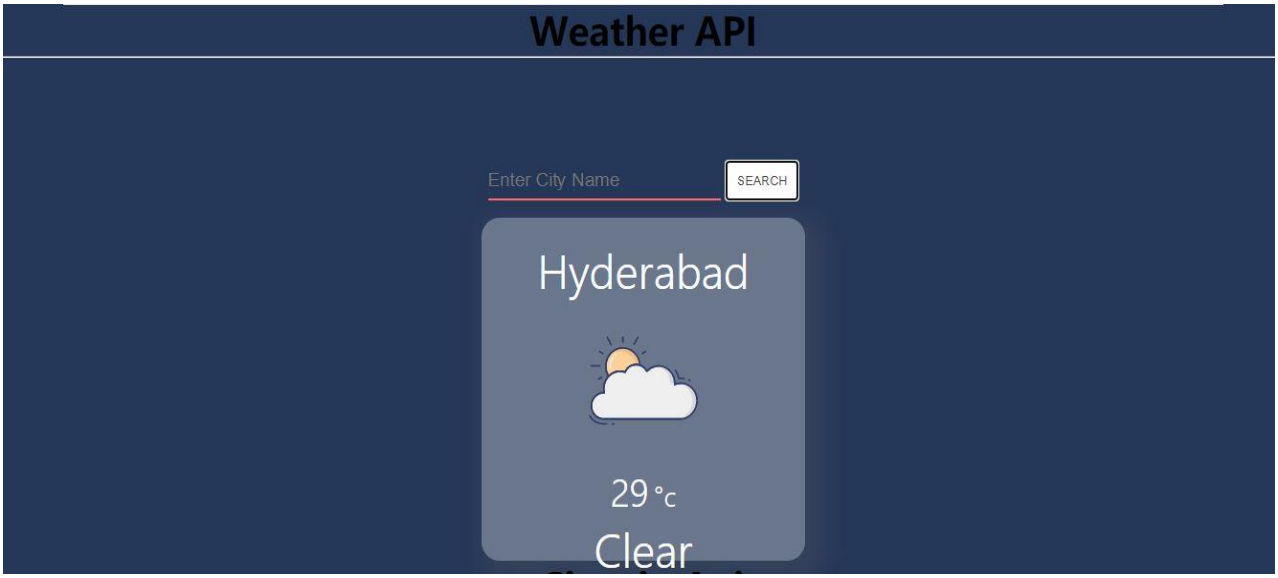
Google Map API Result on web page:



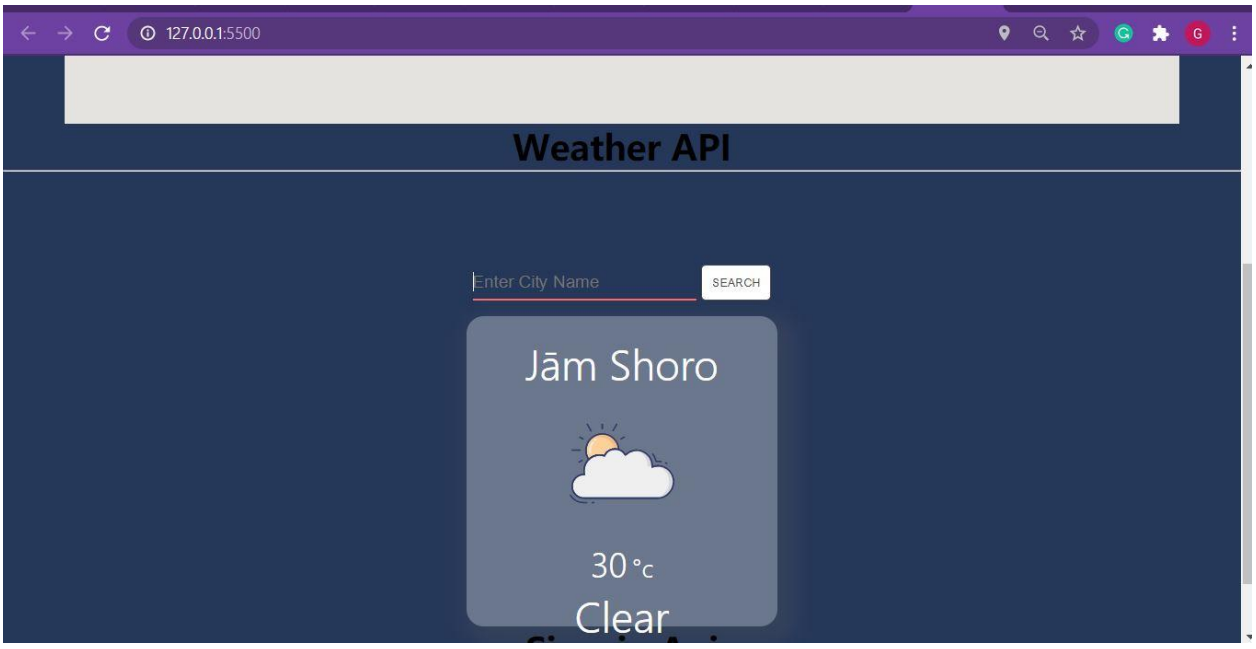
OpenWeather API Result on web page:



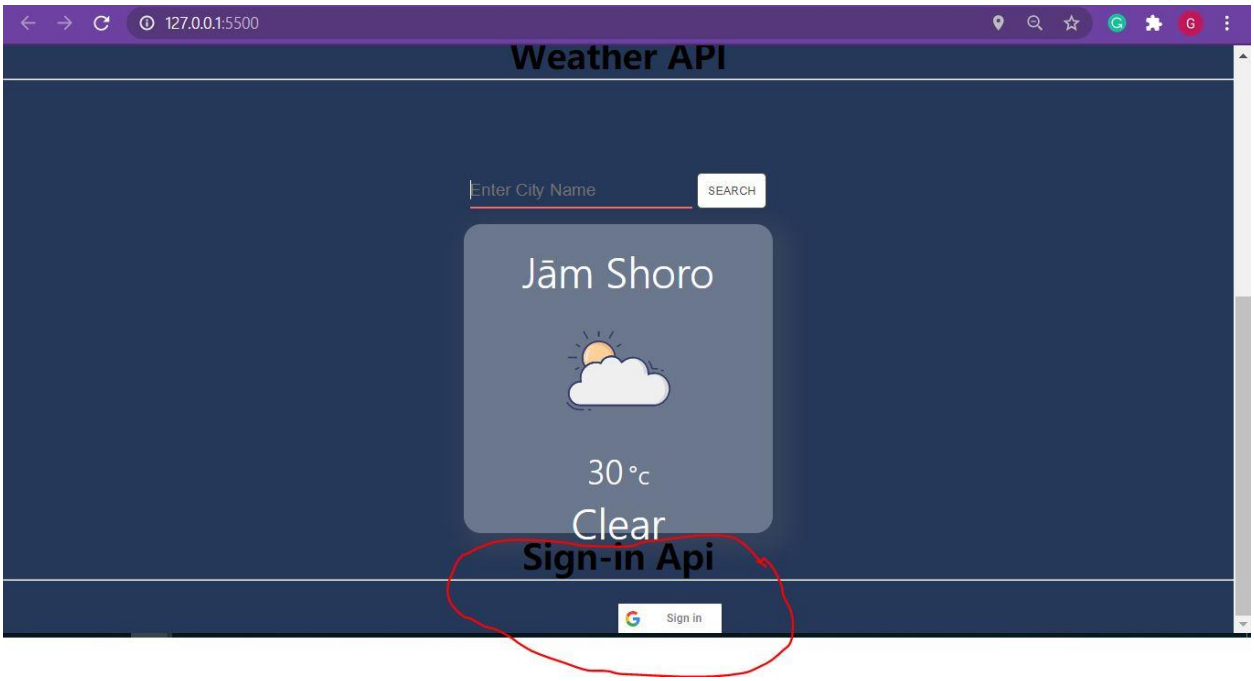
OpenWeather API with “Hyderabad” Search.



OpenWeather API with “jamshoro” Search.



Google Sign-In API Result on web page:



Click on Google Sign-In API Button Result on web page: