**GOOGLE API DOCUMENTATION**

**Chapter -1**

1. **Create an Html Page**

<!DOCTYPE html>  
<html>  
  <head>  
    <style>  
       #map {  
        height: 400px;  
        width: 100%;  
       }  
    </style>  
  </head>  
  <body>  
    <h3>My Google Maps Demo</h3>  
    <div id="map"></div>  
    <script>  
      function initMap() {  
        var uluru = {lat: -25.363, lng: 131.044};  
        var map = new google.maps.Map(document.getElementById('map'), {  
          zoom: 4,  
          center: uluru  
        });  
    }  
    </script>  
    <script async defer  
    src="https://maps.googleapis.com/maps/api/js?key=***YOUR\_API\_KEY***&callback=initMap">  
    </script>  
  </body>  
</html>

**Note:**

<script async defer  
  src="https://maps.googleapis.com/maps/api/js?key=***YOUR\_API\_KEY***&callback=initMap">  
</script>

* The &callback parameter executes the initMap() function after the API loads.
* The async attributes allows the browser to continue rendering the rest of the page while the API loads
* The defer attributes is a Boolean attribute if it is present then it specifies that the script is executed when page has finished the parsing. And it is used for only external script.

1. **Add a map with Marker**

<!DOCTYPE html>  
<html>  
  <head>  
    <style>  
      #map {  
        height: 400px;  
        width: 100%;  
       }  
    </style>  
  </head>  
  <body>  
    <h3>My Google Maps Demo</h3>  
    <div id="map"></div>  
    <script>  
      function initMap() {  
        var uluru = {lat: -25.363, lng: 131.044};  
        var map = new google.maps.Map(document.getElementById('map'), {  
          zoom: 4,  
          center: uluru  
        });  
        var marker = new google.maps.Marker({  
          position: uluru,  
          map: map  
        });  
      }  
    </script>  
    <script async defer  
    src="https://maps.googleapis.com/maps/api/js?key=***YOUR\_API\_KEY***&callback=initMap">  
    </script>  
  </body>  
</html>

**3. Get an API key**

Step1: Go to Google API console.

Step2: Create or Select a Project.

Step3: Click continues to enable the API.

Step4: On the Credentials page, get an API key.

**Chapter -2**

**Direction Google Map API**

The Directions API is a service that calculates directions between locations using an HTTP request.

* 1. Displaying Text direction with setPanel()

<!DOCTYPE html>  
<html>  
  <head>  
    <meta name="viewport" content="initial-scale=1.0, user-scalable=no">  
    <meta charset="utf-8">  
    <title>Displaying text directions with <code>setPanel()</code></title>  
    <style>  
      /\* Always set the map height explicitly to define the size of the div  
       \* element that contains the map. \*/  
      #map {  
        height: 100%;  
      }  
      /\* Optional: Makes the sample page fill the window. \*/  
      html, body {  
        height: 100%;  
        margin: 0;  
        padding: 0;  
      }  
      #floating-panel {  
        position: absolute;  
        top: 10px;  
        left: 25%;  
        z-index: 5;  
        background-color: #fff;  
        padding: 5px;  
        border: 1px solid #999;  
        text-align: center;  
        font-family: 'Roboto','sans-serif';  
        line-height: 30px;  
        padding-left: 10px;  
      }  
      #right-panel {  
        font-family: 'Roboto','sans-serif';  
        line-height: 30px;  
        padding-left: 10px;  
      }  
  
      #right-panel select, #right-panel input {  
        font-size: 15px;  
      }  
  
      #right-panel select {  
        width: 100%;  
      }  
  
      #right-panel i {  
        font-size: 12px;  
      }  
      #right-panel {  
        height: 100%;  
        float: right;  
        width: 390px;  
        overflow: auto;  
      }  
      #map {  
        margin-right: 400px;  
      }  
      #floating-panel {  
        background: #fff;  
        padding: 5px;  
        font-size: 14px;  
        font-family: Arial;  
        border: 1px solid #ccc;  
        box-shadow: 0 2px 2px rgba(33, 33, 33, 0.4);  
        display: none;  
      }  
      @media print {  
        #map {  
          height: 500px;  
          margin: 0;  
        }  
        #right-panel {  
          float: none;  
          width: auto;  
        }  
      }  
    </style>  
  </head>  
  <body>  
    <div id="floating-panel">  
      <strong>Start:</strong>  
      <select id="start">  
        <option value="chicago, il">Chicago</option>  
        <option value="st louis, mo">St Louis</option>  
        <option value="joplin, mo">Joplin, MO</option>  
        <option value="oklahoma city, ok">Oklahoma City</option>  
        <option value="amarillo, tx">Amarillo</option>  
        <option value="gallup, nm">Gallup, NM</option>  
        <option value="flagstaff, az">Flagstaff, AZ</option>  
        <option value="winona, az">Winona</option>  
        <option value="kingman, az">Kingman</option>  
        <option value="barstow, ca">Barstow</option>  
        <option value="san bernardino, ca">San Bernardino</option>  
        <option value="los angeles, ca">Los Angeles</option>  
      </select>  
      <br>  
      <strong>End:</strong>  
      <select id="end">  
        <option value="chicago, il">Chicago</option>  
        <option value="st louis, mo">St Louis</option>  
        <option value="joplin, mo">Joplin, MO</option>  
        <option value="oklahoma city, ok">Oklahoma City</option>  
        <option value="amarillo, tx">Amarillo</option>  
        <option value="gallup, nm">Gallup, NM</option>  
        <option value="flagstaff, az">Flagstaff, AZ</option>  
        <option value="winona, az">Winona</option>  
        <option value="kingman, az">Kingman</option>  
        <option value="barstow, ca">Barstow</option>  
        <option value="san bernardino, ca">San Bernardino</option>  
        <option value="los angeles, ca">Los Angeles</option>  
      </select>  
    </div>  
    <div id="right-panel"></div>  
    <div id="map"></div>  
    <script>  
      function initMap() {  
        var directionsDisplay = new google.maps.DirectionsRenderer;  
        var directionsService = new google.maps.DirectionsService;  
        var map = new google.maps.Map(document.getElementById('map'), {  
          zoom: 7,  
          center: {lat: 41.85, lng: -87.65}  
        });  
        directionsDisplay.setMap(map);  
        directionsDisplay.setPanel(document.getElementById('right-panel'));  
  
        var control = document.getElementById('floating-panel');  
        control.style.display = 'block';  
        map.controls[google.maps.ControlPosition.TOP\_CENTER].push(control);  
  
        var onChangeHandler = function() {  
          calculateAndDisplayRoute(directionsService, directionsDisplay);  
        };  
        document.getElementById('start').addEventListener('change', onChangeHandler);  
        document.getElementById('end').addEventListener('change', onChangeHandler);  
      }  
  
      function calculateAndDisplayRoute(directionsService, directionsDisplay) {  
        var start = document.getElementById('start').value;  
        var end = document.getElementById('end').value;  
        directionsService.route({  
          origin: start,  
          destination: end,  
          travelMode: 'DRIVING'  
        }, function(response, status) {  
          if (status === 'OK') {  
            directionsDisplay.setDirections(response);  
          } else {  
            window.alert('Directions request failed due to ' + status);  
          }  
        });  
      }  
    </script>  
    <script async defer  
    src="https://maps.googleapis.com/maps/api/js?key=*YOUR\_API\_KEY*&callback=initMap">  
    </script>  
  </body>  
</html>

2.1 Travel Mode in Directions:

<!DOCTYPE html>  
<html>  
  <head>  
    <meta name="viewport" content="initial-scale=1.0, user-scalable=no">  
    <meta charset="utf-8">  
    <title>Travel modes in directions</title>  
    <style>  
      /\* Always set the map height explicitly to define the size of the div  
       \* element that contains the map. \*/  
      #map {  
        height: 100%;  
      }  
      /\* Optional: Makes the sample page fill the window. \*/  
      html, body {  
        height: 100%;  
        margin: 0;  
        padding: 0;  
      }  
      #floating-panel {  
        position: absolute;  
        top: 10px;  
        left: 25%;  
        z-index: 5;  
        background-color: #fff;  
        padding: 5px;  
        border: 1px solid #999;  
        text-align: center;  
        font-family: 'Roboto','sans-serif';  
        line-height: 30px;  
        padding-left: 10px;  
      }  
    </style>  
  </head>  
  <body>  
    <div id="floating-panel">  
    <b>Mode of Travel: </b>  
    <select id="mode">  
      <option value="DRIVING">Driving</option>  
      <option value="WALKING">Walking</option>  
      <option value="BICYCLING">Bicycling</option>  
      <option value="TRANSIT">Transit</option>  
    </select>  
    </div>  
    <div id="map"></div>  
    <script>  
      function initMap() {  
        var directionsDisplay = new google.maps.DirectionsRenderer;  
        var directionsService = new google.maps.DirectionsService;  
        var map = new google.maps.Map(document.getElementById('map'), {  
          zoom: 14,  
          center: {lat: 37.77, lng: -122.447}  
        });  
        directionsDisplay.setMap(map);  
  
        calculateAndDisplayRoute(directionsService, directionsDisplay);  
        document.getElementById('mode').addEventListener('change', function() {  
          calculateAndDisplayRoute(directionsService, directionsDisplay);  
        });  
      }  
  
      function calculateAndDisplayRoute(directionsService, directionsDisplay) {  
        var selectedMode = document.getElementById('mode').value;  
        directionsService.route({  
          origin: {lat: 37.77, lng: -122.447},  // Haight.  
          destination: {lat: 37.768, lng: -122.511},  // Ocean Beach.  
          // Note that Javascript allows us to access the constant  
          // using square brackets and a string value as its  
          // "property."  
          travelMode: google.maps.TravelMode[selectedMode]  
        }, function(response, status) {  
          if (status == 'OK') {  
            directionsDisplay.setDirections(response);  
          } else {  
            window.alert('Directions request failed due to ' + status);  
          }  
        });  
      }  
    </script>  
    <script async defer  
    src="https://maps.googleapis.com/maps/api/js?key=*YOUR\_API\_KEY*&callback=initMap">  
    </script>  
  </body>  
</html>

**Chapter – 3**

**Distance Matrix API**

The Distance Matrix API is a service that **provides travel distance and time** for a matrix of origins and destinations.