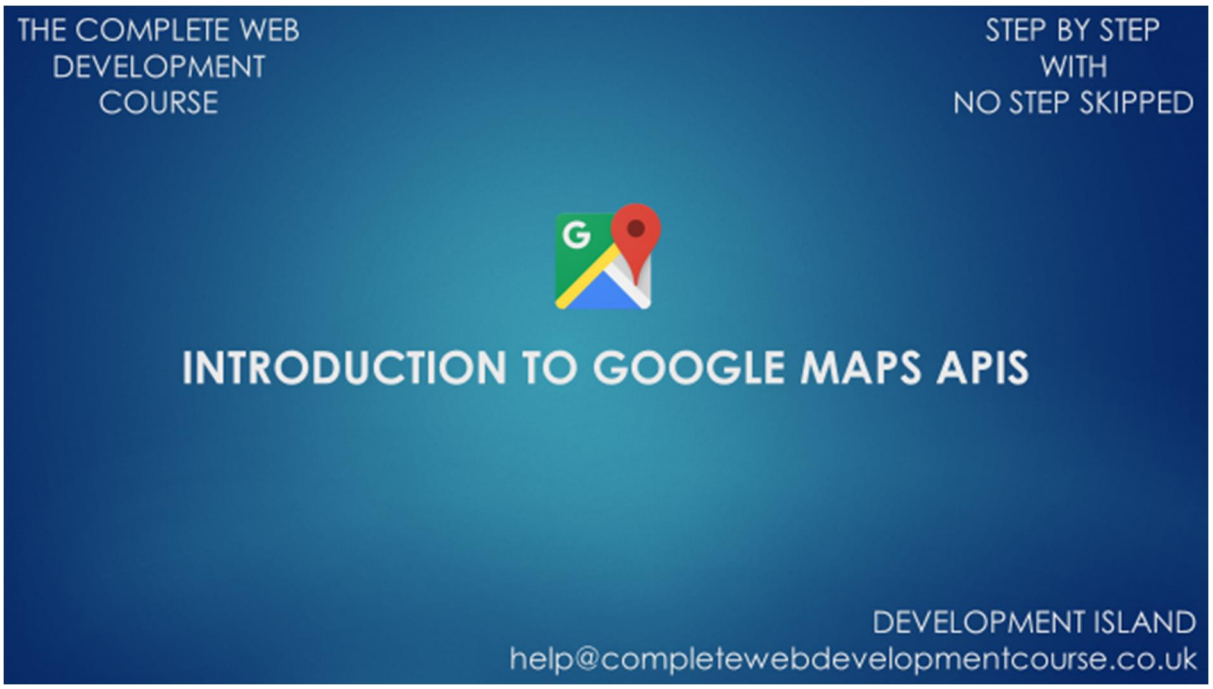
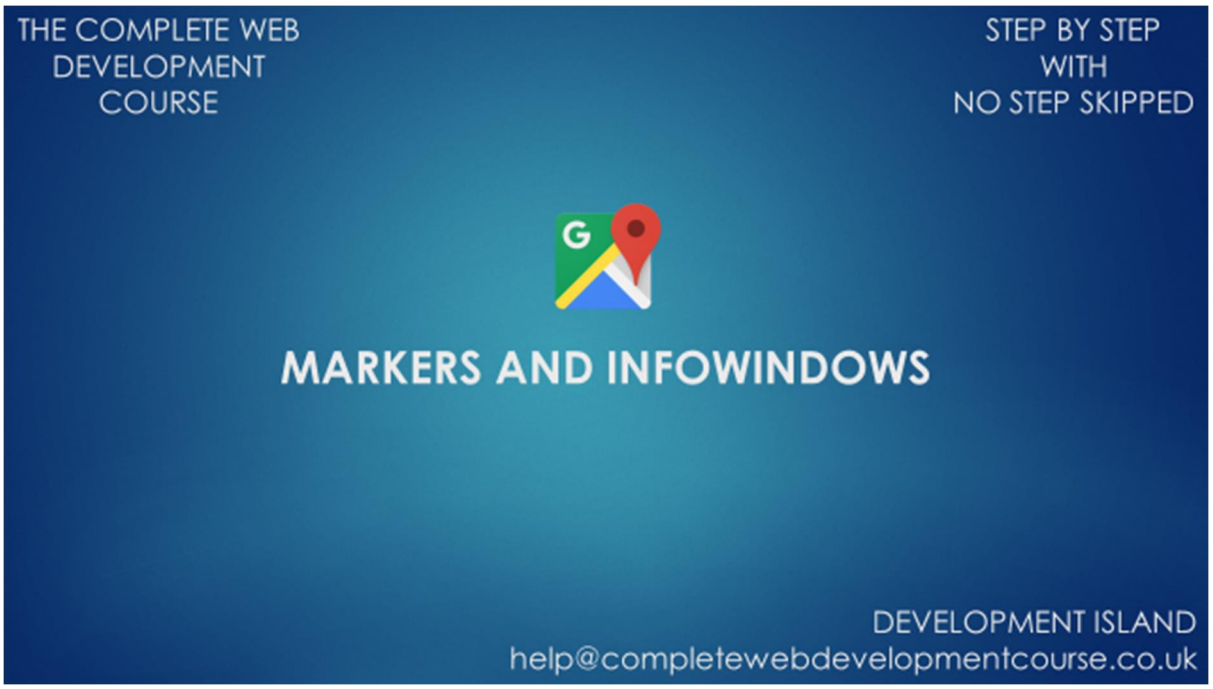



```
<!DOCTYPE html>
<html>
    <head>
        <title>Introduction to Google Maps Javascript
API</title>
        <meta name="viewport" content="initial-scale=1.0,
user-scalable=no">
        <meta charset="utf-8">
        <style>
            html, body{
                height:100%;
            }
            #map{
                height:60%;
            }
        </style>
    </head>

    <body>
        <div id="map"></div>

        <script
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCwJ
2Vepe9L2Miuh7QH87SR_RItIXHlX6Q"></script>
        <script>
```

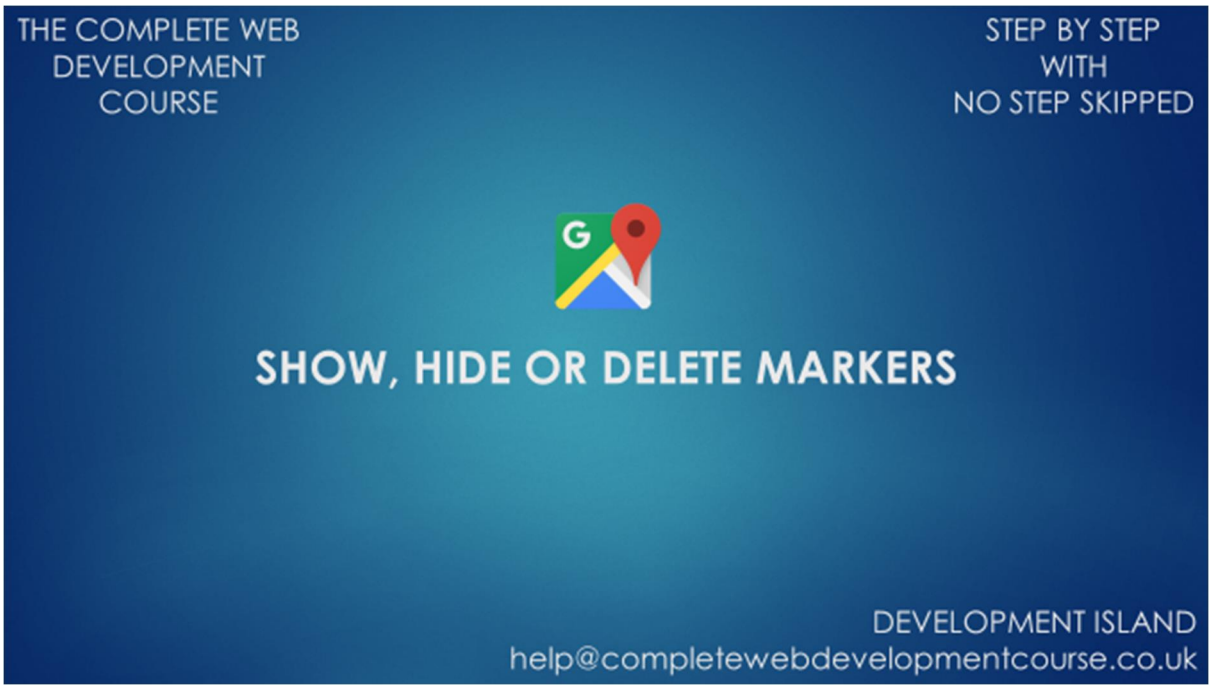
```
            //set map options
            var myLatLng = {lat: 51.5, lng: -0.1};
            var mapOptions = {
                center: myLatLng,
                zoom: 7,
                mapTypeId: google.maps.MapTypeId.SATELLITE

            };

            //create map
            var map = new
google.maps.Map(document.getElementById('map'),
mapOptions);

            //setting the mapTypeId upon construction

map.setMapTypeId(google.maps.MapTypeId.ROADMAP);
        </script>
    </body>

</html>
```




```
<!DOCTYPE html>
<html>
    <head>
        <title>Markers and Infowindows</title>
        <meta name="viewport" content="initial-scale=1.0,
user-scalable=no">
        <meta charset="utf-8">
        <style>
            html, body{
                height:100%;
            }
            #map{
                height:60%;
            }
        </style>
    </head>

    <body>
        <div id="map"></div>

        <script
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCwJ
2Vepe9L2Miuh7QH87SR_RItIXHlX6Q"></script>
        <script>

            //set map options
```

```
            var myLatLng = {lat: 51.5, lng: -0.1};
            var mapOptions = {
                center: myLatLng,
                zoom: 7,
                mapTypeId: google.maps.MapTypeId.SATELLITE

            };

            //create map
            var map = new
google.maps.Map(document.getElementById('map'),
mapOptions);

            //setting the mapTypeId upon construction

map.setMapTypeId(google.maps.MapTypeId.ROADMAP);


            //create marker1
            var marker1Options = {
                position: myLatLng,
                map: map,
                title: "This is London",
                draggable: true,
                animation: google.maps.Animation.DROP
//ANIMATION

            }
            var marker1 = new
google.maps.Marker(marker1Options);

            //create InfoWindow
            var contentString = "<div>This is an
InfoWindow</div>";
            var infowindow = new google.maps.InfoWindow({
                content: contentString,
                maxWidth: 100
            });

            //add lister to the marker to show InfoWindow

            marker1.addListener("click", function(){
```

```
            infowindow.open(map, marker1);
            });




            //create marker2
            var marker2Options = {
                position: {lat:52.1337, lng: -0.4577},
                title: "This is Bedford."
            }
            var marker2 = new
google.maps.Marker(marker2Options);

marker2.setAnimation(google.maps.Animation.BOUNCE);
            marker2.setMap(map);
        </script>
    </body>

</html>
```

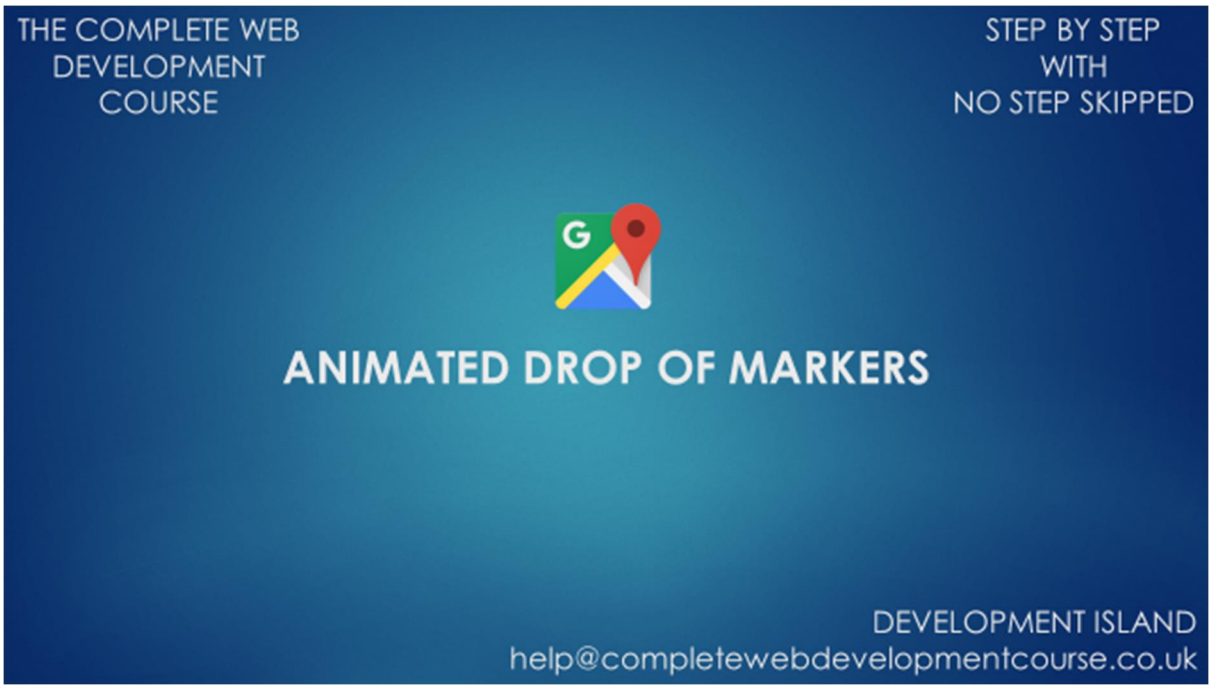



```
<!DOCTYPE html>
<html>
    <head>
        <title>Show, hide, delete markers</title>
        <meta name="viewport" content="initial-scale=1.0,
user-scalable=no">
        <meta charset="utf-8">
        <style>
            html, body{
                height:100%;
            }
            #map{
                height:60%;
            }
        </style>
    </head>

    <body>
        <div id="map"></div>
        <button onclick="showMarkers();">Show Existing
Markers</button>
        <button onclick="clearMarkers();">Hide Markers
from the map</button>
        <button onclick="deleteMarkers();">remove Markers
Completely</button>
```

```
        <script
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCwJ
2Vepe9L2Miuh7QH87SR_RItIXHlX6Q"></script>
        <script>

            //set map options
            var myLatLng = {lat: 51.5, lng: -0.1};
            var mapOptions = {
                center: myLatLng,
                zoom: 7,
                mapTypeId: google.maps.MapTypeId.SATELLITE

            };

            //create map
            var map = new
google.maps.Map(document.getElementById('map'),
mapOptions);

            //setting the mapTypeId upon construction

map.setMapTypeId(google.maps.MapTypeId.ROADMAP);


            //create marker1
            var marker1Options = {
                position: myLatLng,
                map: map,
                title: "This is London",
                draggable: true,
                animation: google.maps.Animation.DROP
//ANIMATION

            }
            var marker1 = new
google.maps.Marker(marker1Options);

            //create InfoWindow
            var contentString = "<div>This is an
InfoWindow</div>";
            var infowindow = new google.maps.InfoWindow({
                content: contentString,
```

```
                maxWidth: 100
            });

            //add lister to the marker to show InfoWindow

            marker1.addListener("click", function(){
            infowindow.open(map, marker1);
            });




            //create marker2
            var marker2Options = {
                position: {lat:52.1337, lng: -0.4577},
                title: "This is Bedford."
            }
            var marker2 = new
google.maps.Marker(marker2Options);

marker2.setAnimation(google.maps.Animation.BOUNCE);
            marker2.setMap(map);

            //remove markers
            marker1.setMap(null);
            marker2.setMap(null);

            //create array where we store markers
            var markers = [];
            //create marker once we click on a point on
the map
            map.addListener("click", function(event){
            var markerOptions = {
                position: event.latLng,
//                map: map
            }
            var marker = new
google.maps.Marker(markerOptions);
                //store marker in array
                markers.push(marker);
            });
```

```
            //show markers stored in the array
            function showMarkers(){
                for(var i=0; i<markers.length; i++){
                    markers[i].setMap(map);
                }
            }

            //hide markers from the map (they are still in
the array)
            function clearMarkers(){
                for(var i=0; i<markers.length; i++){
                    markers[i].setMap(null);
                }
            }

            //delete markers from the array
            function deleteMarkers(){
                clearMarkers();
                markers = [];
            }
        </script>
    </body>

</html>
```

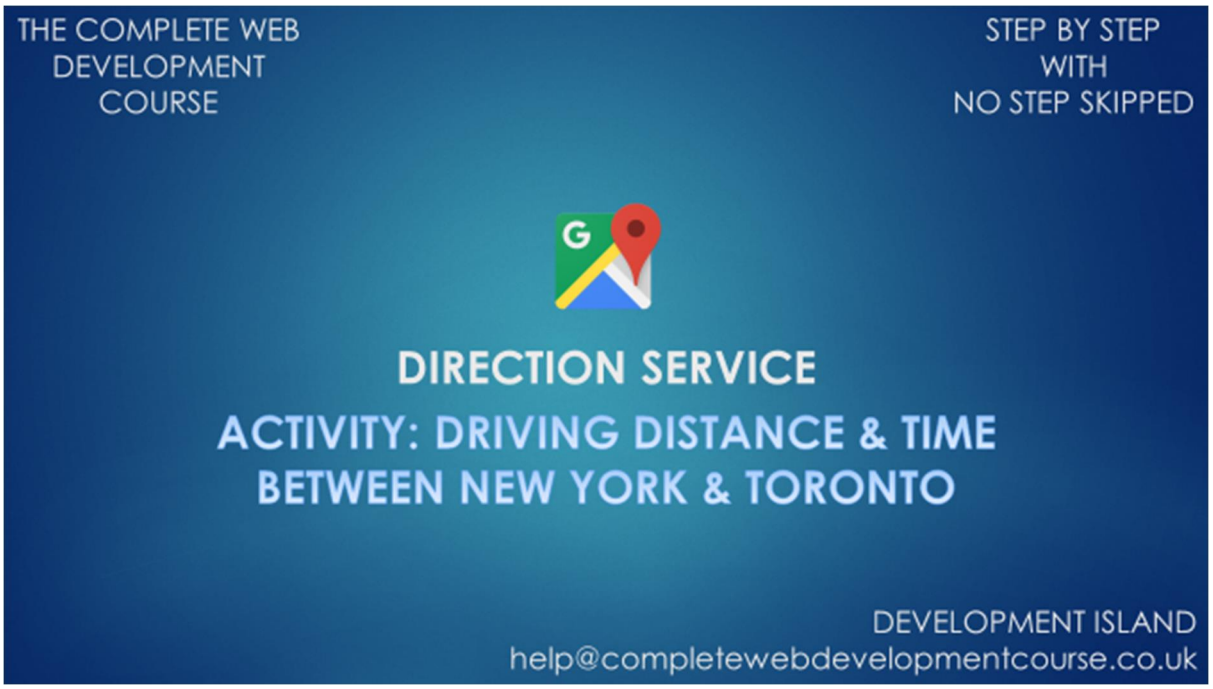



```
<!DOCTYPE html>
<html>
    <head>
        <title>markers animation</title>
        <meta name="viewport" content="initial-scale=1.0,
user-scalable=no">
        <meta charset="utf-8">
        <style>
            html, body{
                height:100%;
            }
            #map{
                height:60%;
            }
        </style>
    </head>

    <body>
        <div id="map"></div>
        <button onclick="showMarkers();">Show Existing
Markers</button>
        <button onclick="clearMarkers();">Hide Markers
from the map</button>
        <button onclick="deleteMarkers();">remove Markers
Completely</button>
```

```
        <script
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCwJ
2Vepe9L2Miuh7QH87SR_RItIXHlX6Q"></script>
        <script>

            //set map options
            var myLatLng = {lat: 51.5, lng: -0.1};
            var mapOptions = {
                center: myLatLng,
                zoom: 7,
                mapTypeId: google.maps.MapTypeId.SATELLITE

            };

            //create map
            var map = new
google.maps.Map(document.getElementById('map'),
mapOptions);

            //setting the mapTypeId upon construction

map.setMapTypeId(google.maps.MapTypeId.ROADMAP);


            //create marker1
            var marker1Options = {
                position: myLatLng,
                map: map,
                title: "This is London",
                draggable: true,
                animation: google.maps.Animation.DROP
//ANIMATION

            }
            var marker1 = new
google.maps.Marker(marker1Options);

            //create InfoWindow
            var contentString = "<div>This is an
InfoWindow</div>";
            var infowindow = new google.maps.InfoWindow({
                content: contentString,
```

```
                maxWidth: 100
            });

            //add lister to the marker to show InfoWindow

            marker1.addListener("click", function(){
            infowindow.open(map, marker1);
            });




            //create marker2
            var marker2Options = {
                position: {lat:52.1337, lng: -0.4577},
                title: "This is Bedford."
            }
            var marker2 = new
google.maps.Marker(marker2Options);

marker2.setAnimation(google.maps.Animation.BOUNCE);
            marker2.setMap(map);

            //remove markers
            marker1.setMap(null);
            marker2.setMap(null);

            //create array where we store markers
            var markers = [];
            //create marker once we click on a point on
the map
            map.addListener("click", function(event){
            var markerOptions = {
                position: event.latLng,
//                map: map
            }
            var marker = new
google.maps.Marker(markerOptions);
                //store marker in array
                markers.push(marker);
            });
```

```
            //show markers stored in the array
            function showMarkers(){
                for(var i=0; i<markers.length; i++){
                    addMarkerwithDelay(i);
                }
            }

            function addMarkerwithDelay(i){
                setTimeout(function(){
                    markers[i].setMap(map);

markers[i].setAnimation(google.maps.Animation.DROP);

                }, 200*i);
            }

            //hide markers from the map (they are still in
the array)
            function clearMarkers(){
                for(var i=0; i<markers.length; i++){
                    markers[i].setMap(null);
                }
            }

            //delete markers from the array
            function deleteMarkers(){
                clearMarkers();
                markers = [];
            }
        </script>
    </body>

</html>
```




```
<!DOCTYPE html>
<html>
    <head>
        <title>Direction service</title>
        <meta name="viewport" content="initial-scale=1.0,
user-scalable=no">
        <meta charset="utf-8">
        <style>
            html, body{
                height:100%;
            }
            #map{
                height:60%;
            }
        </style>
    </head>

    <body>
        <div id="map"></div>
        <button onclick="calcRoute();">Calculate
Route</button>
        <script
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCwJ
2Vepe9L2Miuh7QH87SR_RItIXHlX6Q"></script>
        <script>
```

```
            //set map options
            var myLatLng = {lat: 51.5, lng: -0.1};
            var mapOptions = {
                center: myLatLng,
                zoom: 7,
                mapTypeId: google.maps.MapTypeId.ROADMAP

            };

            //create map
            var map = new
google.maps.Map(document.getElementById('map'),
mapOptions);

            //create a DirectionsService object to use the
route method and get a result for our request
            var directionsService = new
google.maps.DirectionsService();

            //create a DirectionsRenderer object which we
will use to display the route
            var directionsDisplay = new
google.maps.DirectionsRenderer();

            //bind the DirectionsRenderer to the map
            directionsDisplay.setMap(map);

            //define calcRoute function
            function calcRoute(){
                var request = {
                    origin: "New York",
                    destination: "Toronto",
                    travelMode:
google.maps.TravelMode.DRIVING, //WALKING, BYCYCLING,
TRANSIT
                    unitSystem:
google.maps.UnitSystem.METRIC
                }

                //pass the request to the route method
                directionsService.route(request,
function(result, status){
```

```
                if(status ==
google.maps.DirectionsStatus.OK){
                    console.log(result);

                    //Get distance and time
                    window.alert("The travelling distance
is " + result.routes[0].legs[0].distance.text + ".<br
/>The travelling time is: " +
result.routes[0].legs[0].duration.text + ".");

                    //display route

directionsDisplay.setDirections(result);
                }
                });
            }

        </script>
    </body>

</html>
```


```
<!DOCTYPE html>
<html>
    <head>
        <title>Geocoding using Google Maps Javascript
API</title>
        <meta name="viewport" content="initial-scale=1.0,
user-scalable=no">
        <meta charset="utf-8">
        <style>
            html, body{
                height:100%;
            }
            #map{
                height:60%;
            }
        </style>
    </head>

    <body>
        <div id="map"></div>
        <input type="text" placeholder="Address"
id="address">
        <button onclick="geocodeAddress();">geocode
Addess</button>
```

```
        <script
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCwJ
2Vepe9L2Miuh7QH87SR_RItIXHlX6Q"></script>
        <script>

            //set map options
            var myLatLng = {lat: 51.5, lng: -0.1};
            var mapOptions = {
                center: myLatLng,
                zoom: 7,
                mapTypeId: google.maps.MapTypeId.ROADMAP

            };

            //create map
            var map = new
google.maps.Map(document.getElementById('map'),
mapOptions);
            //create a geocoder object to use the geocode
method
            var geocoder = new google.maps.Geocoder();


            //create geocode function
            function geocodeAddress(){
                geocoder.geocode({'address':
document.getElementById("address").value},
function(results, status){
                    if(status ==
google.maps.GeocoderStatus.OK){
                        console.log(results);
                        window.alert("Address coordinates:
" + results[0].geometry.location);

map.setCenter(results[0].geometry.location);
                        var marker = new
google.maps.Marker({
                            map: map,
                            position:
results[0].geometry.location
                        });
                    }else{
```

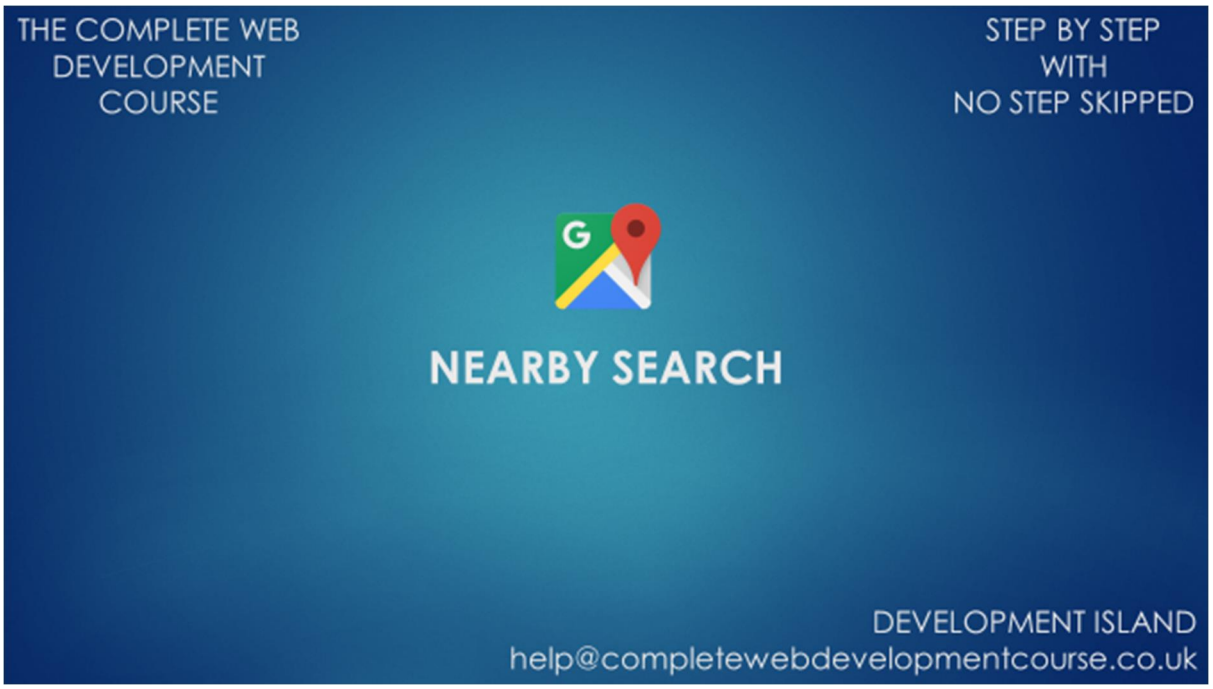
```
                        window.alert("Geocode not
successful: " + status);
                    }
                });
            }
        </script>
    </body>

</html>
```


```
<!DOCTYPE html>
<html>
    <head>
        <title>Geocoding API</title>
        <meta name="viewport" content="initial-scale=1.0,
user-scalable=no">
        <meta charset="utf-8">
        <style>
            html, body{
                height:100%;
            }
            #map{
                height:60%;
            }
        </style>
    </head>

    <body>
        <div id="map"></div>
        <input type="text" placeholder="Address"
id="address">
        <button onclick="geocodeAddress();">geocode
Addess</button>
        <div id="output"></div>
```

```
        <script
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCwJ
2Vepe9L2Miuh7QH87SR_RItIXHlX6Q"></script>
        <script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/j
query.min.js"></script>
        <script>

            //set map options
            var myLatLng = {lat: 51.5, lng: -0.1};
            var mapOptions = {
                center: myLatLng,
                zoom: 7,
                mapTypeId: google.maps.MapTypeId.ROADMAP

            };

            //create map
            var map = new
google.maps.Map(document.getElementById('map'),
mapOptions);


            //define marker variable
            var marker;
            //geocode function
            function geocodeAddress(){
                var url =
"https://maps.googleapis.com/maps/api/geocode/json?address
="+document.getElementById("address").value+"&key=AIzaSyCw
J2Vepe9L2Miuh7QH87SR_RItIXHlX6Q";
//                window.open(url);

                $.getJSON(url, function(data){
                    if(data.status == "OK"){
                        var formattedAddress =
data.results[0].formatted_address;
                        var latitude =
data.results[0].geometry.location.lat;
                        var longitude =
data.results[0].geometry.location.lng;
                        var postcode;
```

```
$.each(data.results[0].address_components, function(index,
element){
                        if(element.types ==
"postal_code"){
                            postcode = element.long_name;
                            return false; //stop the loop
                        }
                        });

                        $("#output").html("<b>Formatted
Address</b>:" + formattedAddress + ".<br
/><b>Coordinates</b>: (lat: " + latitude + ", lng: " +
longitude + ").<br /><b>Postcode</b>: " + postcode + ".");

                        //center map
                        map.setCenter({lat:
latitude,lng:longitude});
                        //change zoom level
                        map.setZoom(14);

                        //if marker is there delete it
                        if(marker != undefined){
                            marker.setMap(null);
                        }
                        //create marker
                        marker = new google.maps.Marker({
                            map: map,
                            position: {lat:
latitude,lng:longitude},
                            animation:
google.maps.Animation.DROP
                        });

                    }else{
                        $("#output").html("Request
unsuccessful");
                    }
                });

            }
```

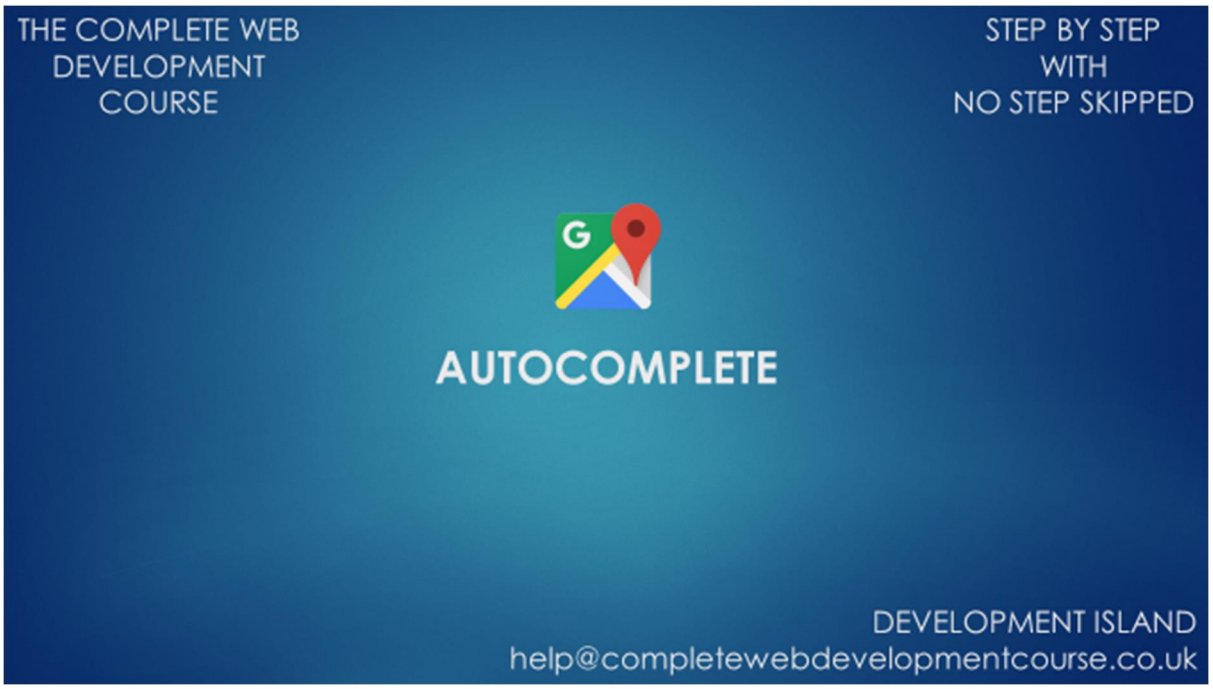
```
        </script>
    </body>

</html>
```




```
<!DOCTYPE html>
<html>
    <head>
        <title>nearby search</title>
        <meta name="viewport" content="initial-scale=1.0,
user-scalable=no">
        <meta charset="utf-8">
        <style>
            html, body{
                height:100%;
            }
            #map{
                height:60%;
            }
        </style>
    </head>

    <body>
        <div id="map"></div>
        <script
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCwJ
2Vepe9L2Miuh7QH87SR_RItIXHlX6Q&libraries=places"></script>
        <script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/j
query.min.js"></script>
        <script>
```

```
            //set map options
            var myLatLng = {lat: 51.5, lng: -0.1};
            var mapOptions = {
                center: myLatLng,
                zoom: 15,
                mapTypeId: google.maps.MapTypeId.ROADMAP

            };

            //create map
            var map = new
google.maps.Map(document.getElementById('map'),
mapOptions);

            //create infowindow
            var infowindow = new google.maps.InfoWindow();

            //define a request, the location must be
defined using google.maps.LatLng

            var london = new google.maps.LatLng(51.5, -
0.1);
            var request = {
                location: london,
                radius: '500',
                types: ['store']
            }

            //create a placesServcie object before using
the nearbysearch method
            var service = new
google.maps.places.PlacesService(map);

            service.nearbySearch(request, callback);

            //define the callback function showing what we
do with the results

            function callback(result, status){
                if(status ==
google.maps.places.PlacesServiceStatus.OK){
```

```
                    console.log(result);
                    for(i =0; i<result.length; i++){
                        addMarker(result[i]);
                    }
                }
            }

            //add a marker for each place in the result
array
            function addMarker(place){
                   var marker = new google.maps.Marker({
                    map: map,
                       position: place.geometry.location,
                       animation:
google.maps.Animation.DROP
                    });
                google.maps.event.addListener(marker,
"click", function(){
            infowindow.setContent(place.name);
                    infowindow.open(map, this);
                })
            }
        </script>
    </body>

</html>
```




```
<!DOCTYPE html>
<html>
    <head>
        <title>Autocomplete</title>
        <meta name="viewport" content="initial-scale=1.0,
user-scalable=no">
        <meta charset="utf-8">
        <style>
            html, body{
                height:100%;
            }
            #map{
                height:60%;
            }
        </style>
    </head>

    <body>
        <div id="map"></div>
        <input type="text" id="cityInput"
placeholder="City">
        <script
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCwJ
2Vepe9L2Miuh7QH87SR_RItIXHlX6Q&libraries=places"></script>
```

```
        <script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/j
query.min.js"></script>
        <script>

            //set map options
            var myLatLng = {lat: 51.5, lng: -0.1};
            var mapOptions = {
                center: myLatLng,
                zoom: 7,
                mapTypeId: google.maps.MapTypeId.ROADMAP

            };

            //create map
            var map = new
google.maps.Map(document.getElementById('map'),
mapOptions);

            //create autocomplete object
            var input =
document.getElementById("cityInput");
            var options = {
                types: ['(cities)']
            }
            var autocomplete = new
google.maps.places.Autocomplete(input, options);


autocomplete.addListener('place_changed',onPlaceChanged);
            function onPlaceChanged(){

                var place = autocomplete.getPlace();
                map.panTo(place.geometry.location);
            }
        </script>
    </body>

</html>
```