

Nombre: Magallanes Ramírez Diego Alberto.
No. Cuenta: 311343591
Práctica 2, parte 1.

1. ¿En qué consiste la API de google maps?

Nos ofrece las bibliotecas necesarias para poder utilizar los mapas de google maps dentro de nuestras aplicaciones y poder tener acceso a la base de datos mundial de Google.

2. ¿Cómo están expresadas las coordenadas?

Las coordenadas se expresan por dos valores decimales separados por una coma, uno será la latitud y el otro la longitud.

3. ¿Cuánto es el máximo de decimales de una coordenada?

El máximo de decimales que podemos usar son 6. Pero con 2 decimales basta para centrar un país, 3 para una ciudad y a partir de 4 se pueden conseguir mas detalles.

4. ¿Cómo se crea un nuevo mapa y que propiedades puede tener?

1. Antes que nada debemos obtener una clave para poder hacer uso de la API de Google Maps.
2. Una vez obtenida nuestra clave, vamos a nuestro archivo html y en el cuerpo escribimos la siguiente etiqueta:

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

Esta etiqueta reserva un lugar para el mapa, el identificador que se le asigna será con el que se cree el mapa.

3. Ahora dentro del mismo cuerpo creamos una función en Javascript, la cual inicializará el mapa y lo desplegará.

En esta función primero se creara un objeto de la clase Map utilizando el identificador del punto anterior, luego se asignaran los datos con los que queremos que comience nuestro mapa, por ejemplo, las coordenadas donde vera nuestro mapa, el zoom, el tipo de mapa, marcadores, ventanas de información, etc.

```

<script type="text/javascript">
var map;
function initMap(){
    map = new google.maps.Map(document.getElementById('map'), {
        center: {lat: 35.141432, lng: -90.052629},
        zoom: 21,
        mapTypeId: google.maps.MapTypeId.SATELLITE
    });
}
</script>

```

- Ahora solo falta especificar la URL donde se encontrará el archivo de JavaScript que carga todos los símbolos y definiciones para poder usar el API de Google Maps.

```

<script type="text/javascript"
src="https://maps.googleapis.com/maps/api/js?
key=AlzaSyDvfkQSaSfeL20ShaYbhIFZd9n63xAxqSg&sensor=false&callback=initMap">
</script>

```

Los parámetros en la URL son:

- **key:** Indica la clave de uso de la API.
- **sensor:** Indica si la aplicación esta usando o no un dispositivo para detectar la posición del usuario.
- **callback:** llama a la función initMap.

Además, dentro del head, podemos especificar el tamaño que tendrá nuestro mapa, solo debemos agregar las siguientes lineas:

```

<style type="text/css">
#map {height: 100%; margin-left: 25%; margin-right: 25%; margin-top: 2%;
margin-bottom: 2% }
</style>

```

Entre las propiedades que podemos agregar al mapa se encuentran:

- **Tipo de mapa:** Permite cambiar el tipo de mapa a mostrar (rutas, tierra, satélite, híbrido).

- **Control de mapa:** Añade controles de movimiento.
- **Control de escala:** Muestra un elemento de escala del mapa.
- **Control de zoom:** Añade controles de zoom.
- **Control de rotación:** Proporciona una combinación de opciones de inclinación y rotación para mapas que contienen imágenes oblicuas.
- **Control Street View:** Permite habilitar Street View (vista desde la calle).

Para aplicarlas solo debemos escribir en la función initMap:

```
zoomControl: boolean
mapTypeControl: boolean
scaleControl: boolean
streetViewControl: boolean
rotateControl: boolean
```

6. ¿Qué es un marcador?

Los marcadores identifican (marcan) una ubicación específica dentro del mapa por medio de un icono (estos iconos pueden ser fijos o movibles).

7. ¿Cómo se crea un nuevo marcador y que propiedades puede tener?

Para agregar un marcador debemos agregar a la función initMap:

```
var marker = new google.maps.Marker({
  position: {lat: 35.141432, lng: -90.052629},
  map: map,
});
```

Entre las propiedades que podemos agregar al marcador se encuentran:

- Añadir ventana de información.
- Cambiar el icono predeterminado.
- Animar el icono.
- Poder arrastrar el icono.

9. ¿Qué es una ventana de información?

Despliega una etiqueta (estática o dinámica) sobre el marcador donde podemos mostrar información o imágenes que queramos que vea el usuario.

Y solo basta con agregar a la función initMap:

```
var infowindow = new google.maps.InfoWindow({  
  content: info  
});
```

Se le pueden agregar algunos detalles, como que solo se muestre cuando se le de click sobre el marcador.