**Map Component Documentation**

**Introduction**

The Map component is a React functional component responsible for displaying a map using the React Leaflet library. It includes features such as draggable markers and handling map click events.

**Usage**

To use the Map component, import it into your React application and render it in the desired location where you want to display the map.

import Map from './Map';

<Map/>

**Features**

1. **Map Display**: The Map component displays a map using the Leaflet library, allowing users to interact with the map, zoom in/out, and pan across different locations.
2. **Tile Layer**: It includes a tile layer that provides the base map tiles from OpenStreetMap.
3. **GeoJSON Overlay**: The component renders GeoJSON data as an overlay on the map, allowing you to visualize geographical features such as boundaries, regions, or routes.
4. **Draggable Marker**: It includes a draggable marker that users can move around the map to select a specific location.
5. **Map Click Handling**: The component handles click events on the map by providing a callback function (**handleMapClick**) that is triggered when the user clicks on the map. This can be used to capture the coordinates of the clicked location.

**Props**

* **handleMapClick**: A function that handles the click event on the map. It receives the click event object as a parameter and can be used to perform actions based on the clicked location.

**Dependencies**

The Map component relies on the following external libraries:

* react-leaflet: For integrating Leaflet maps into React applications.
* Leaflet: For displaying interactive maps.
* OpenStreetMap: For providing map tiles used as the base layer.

Ensure that these dependencies are installed in your project before using the Map component.

**Important Notes**

* Customize the map center (**center** prop) and initial zoom level (**zoom** prop) based on your application's requirements and the geographical area you want to display.
* Update the URL used for the tile layer to match your preferred tile provider or customize the map's appearance.
* Adjust the GeoJSON data (**data** prop) to include the geographical features you want to overlay on the map.