

LeafLet Mapbox

Obyektif:

- Menyalin access-token dari situs MapBox
- Menyiapkan CSS dan JS files dari Leaflet
- Menempatkan peta di halaman web
- Menciptakan marker, circle dan polygon di peta
- Menggunakan event 'click' untuk mengakses marker dan circle

Daftar Isi

Leaflet Mapbox.....	1
Access-Token	1
Latitude dan Longitude	1
Leaflet	3
Markers, circles and polygons	4
Popup	5
Mengendalikan Event	6

Leaflet Mapbox

Menampilkan peta (map) di Web merupakan hal yang relatif mudah bila menggunakan library atau services dari pihak ketiga seperti dalam hal ini adalah Leaflet. Library ini disiapkan untuk mengakses Mapbox openstreet, sehingga akses program dilakukan dengan cara yang relatif lebih mudah ketimbang harus memprogramkan dari nol. Penggunaan software ini adalah gratis (lihat batasan penggunaan software pada <https://leafletjs.com>)

Access-Token

Untuk dapat mengakses peta di MapBox diperlukan 'access-token' yang harus diambil dari situs Mapbox. Untuk mudahnya dapat di-google "mapbox access token" untuk dapat langsung ke halaman token tersebut. Bila belum mempunyai account, registrasi account baru dengan password.



Salin access-token tersebut ke file editor. **Access-token** ini akan dibutuhkan dalam program untuk menampilkan map openstreet tersebut.

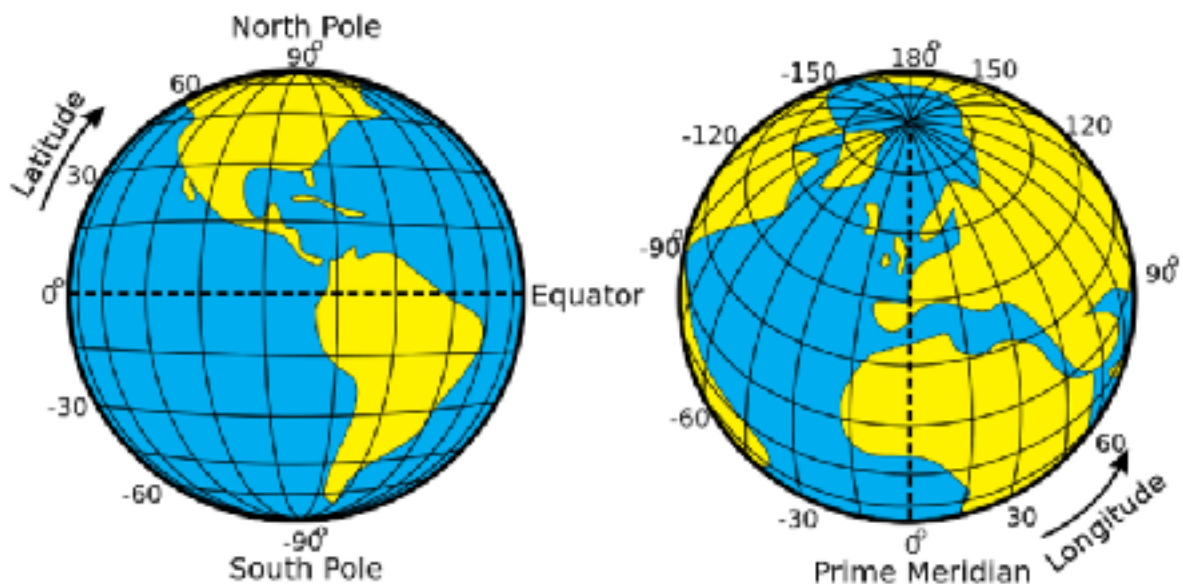
Latitude dan Longitude

Adalah unit yang merepresentasikan koordiant pada sistem geografi, merupakan identitas sebuah lokasi seperti juga alamat sebuah jalan. Setiap tempat diatas bumi dapat dispesifikasikan dengan koordinat latitude dan longitude.

Kedua latitude dan longitude diukur dalam derajat (degree) yang kemudian dibagi dalam menit dan detik.

Identifikasikan angka latitude dan logitude, apakah positif atau negatif. Untuk latitude, baris utara **equator** adalah positif, sedangkan baris selatan adalah negatif. Sedangkan untuk longitude bagian timur **prime meridian** adalah positif, sedangkan bagian barat adalah negatif.

- Contoh: latitude 15.23456 adalah bagian utara dari equator, sedangkan -15.23456 adalah bagian selatan equator.
- Contoh: longitude 30.67890 adalah timur prime meridian dan -30.67890 adalah sebelah barat.



sumber: <https://www.ubergizmo.com/how-to/read-gps-coordinates/>

Untuk melihat koordinat dari Google Map, klik kanan pada map, kemudian pilih submenu "what's here", setelah dipilih, info lokasi akan muncul.



Leaflet

Leaflet adalah library yang mempermudah akses ke Mapbox untuk mendapatkan openstreet map. Persiapkan File CSS dari Leaflet beserta dengan JavaScript:

```
<link rel="stylesheet" href="https://unpkg.com/leaflet@1.3.4/dist/leaflet.css" integrity="sha512-puBpdR07980ZvTTbP4A8Ix/1+A4dHDD0DGqYW6RQ+9jxkRFclaxxQb/SJAWZfWakuyeQUytO7+7N4QKrDh+drA==" crossorigin="" />
```

JavaScript (diletakkan setelah CSS).

```
<script src="https://unpkg.com/leaflet@1.3.4/dist/leaflet.js" integrity="sha512-nMMmRyTVoLYqjP9hrbed9S+FzjZHW5gY1TWCHA5ckwXZBadntCNs8kEqAWdrb907rxbaA44lKTIWjDXZxflOcA==" crossorigin=""></script>
```

Siapkan style untuk penempatan map, dengan tinggi 'fixed' dalam pixel.

```
<style>
  #mapid { height: 180px; }
</style>
```

Pada body html, buat <div> untuk penempatan peta yang akan dibaca.

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



MAP

Tentukan koordinat longitude dan langitude , dalam contoh ini diambil koordinat Legian, sebuah daerah di Bali. Dari Google Map diketahui bahwa koordinatnya adalah :

`[-8.701660,115.169856]`

Inisialisasi dengan zoom-level 13, dan letakkan map tersebut pada elemen '#mapid'.

```
var mymap = L.map('mapid').setView([-8.701660,115.169856], 13);
```

Metode `setView` memberikan nilai balik berupa obyek map.

Menciptakan tile layer memerlukan setting template URL untuk penempatan peta (dalam bentuk image), teks atribut dan maximum level zoom dari layer tersebut.

```
L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?access_token={accessToken}', {
  attribution: 'Map data &copy; <a href="https://www.openstreetmap.org/">OpenStreetMap</a> contributors, <a href="https://creativecommons.org/licenses/by-sa/2.0/">CC-BY-SA</a>, Imagery © <a href="https://www.mapbox.com/">Mapbox</a>',
  maxZoom: 18,
  id: 'mapbox.streets',
  accessToken: 'your.mapbox.access.token'
}).addTo(mymap);
```

Ganti 'your.mapbox.access.token' dengan token yang sudah disimpan dalam editor teks sebelumnya.

URL template

String berikut:

```
'http://{s}.somedomain.com/blabla/{z}/{x}/{y}{r}.png'
```

{s} berarti menampilkan subdomain yang ada, digunakan untuk proses paralel dengan request sesuai limitasi domain.

Nilai sebagai opsi; a, b atau c secara default dapat diabaikan.

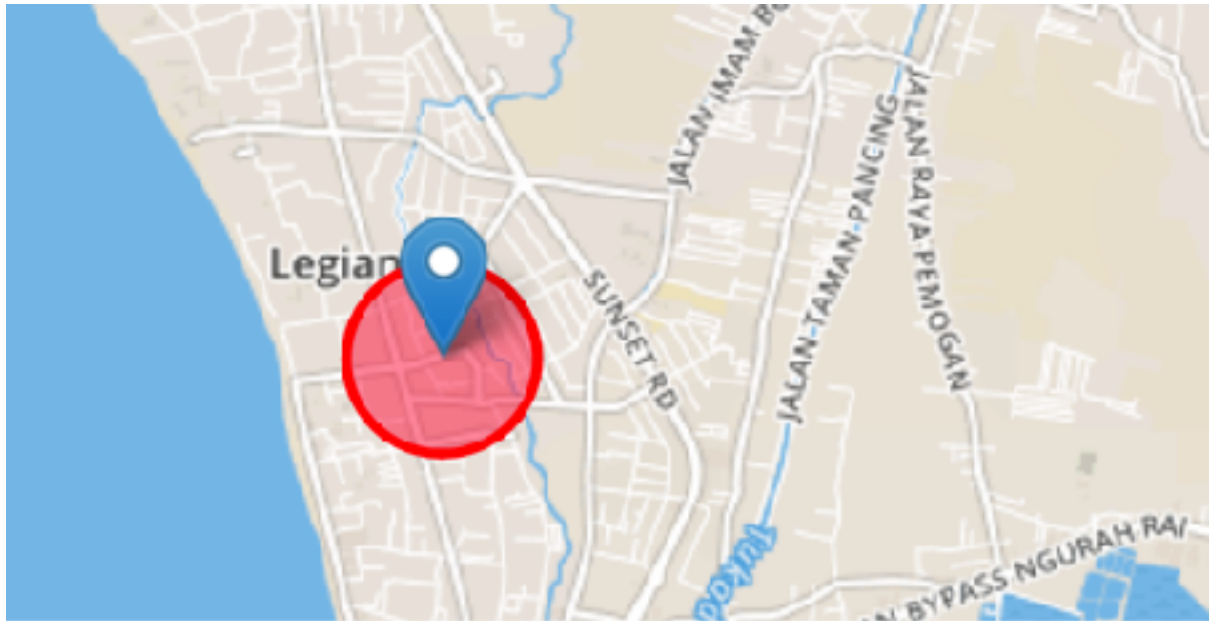
{z} – zoom level, {x} dan {y} – adalah koordinat. {r} bisa digunakan untuk menambahkan "@2x" ke URL untuk mengambil retina tiles. Y

Markers, circles and polygons

Selain lapisan awal map, bisa ditambahkan marker, polylines, polygons, circles, and popups. Contoh membuat marker:

```
var marker = L.marker([-8.701660,115.169856]).addTo(mymap);
```

Cara yang sama digunakan untuk menambahkan circle (kecuali ada radius dalam meter pada argumen kedua).



```
var circle = L.circle([-8.701660,115.169856], {
  color: 'red',
  fillColor: '#f03',
  fillOpacity: 0.5,
  radius: 500
}).addTo(mymap);
```

Atau menambah polygon:

```
var polygon = L.polygon([
  [51.509, -0.08],
  [51.503, -0.06],
  [51.51, -0.047]
]).addTo(mymap);
```

Popup

Popup digunakan sebagai informasi yang muncul pada saat marker atau circle dipilih melalui mouse-klik.

Info bisa didapat langsung pada saat inisialisasi dengan menggunakan metode `openPopup()`.

```
marker.bindPopup("<b>Resto top disini/b><br>yes!").openPopup();
```

```
circle.bindPopup("Ini sebuah circle.");  
polygon.bindPopup("Ini sebuah polygon.");
```

Metode `bindPopup()` menyiapkan popup dengan konten dalam format HTML.

Popup bisa juga dilakukan dengan cara berikut:

```
var popup = L.popup()  
    .setLatLng([-8.703315, 115.168869])  
    .setContent("Restoran terbaik disini.")  
    .openOn(mymap);
```

Metode `openOn` (bukan `addTo`) secara otomatis melakukan 'closing' dari popup sebelumnya yang (mungkin) terbuka.

Mengendalikan Event

Event yang dimaksud adalah klik-mouse pada marker, atau perubahan level zoom. Setiap kali hal tersebut dilakukan, maka event tersebut akan mentrigger obyek untuk menjalankan program tertentu (callback program).

```
function onMapClick(e) {  
    console.log("Peta diklik pada posisi " + e.latlng);  
}
```

```
mymap.on('click', onMapClick);
```

Event ini akan menyampaikan pesan melalui `console.log` melaporkan posisi lokasi yang dipilih (diklik). Program berikut menampilkan popup ketimbang `console.log`.

```
var popup = L.popup();  
  
function onMapClick(e) {  
    popup  
        .setLatLng(e.latlng)  
        .setContent("Lokasi yang dipilih: " + e.latlng.toString())  
        .openOn(mymap);  
}  
  
mymap.on('click', onMapClick);
```