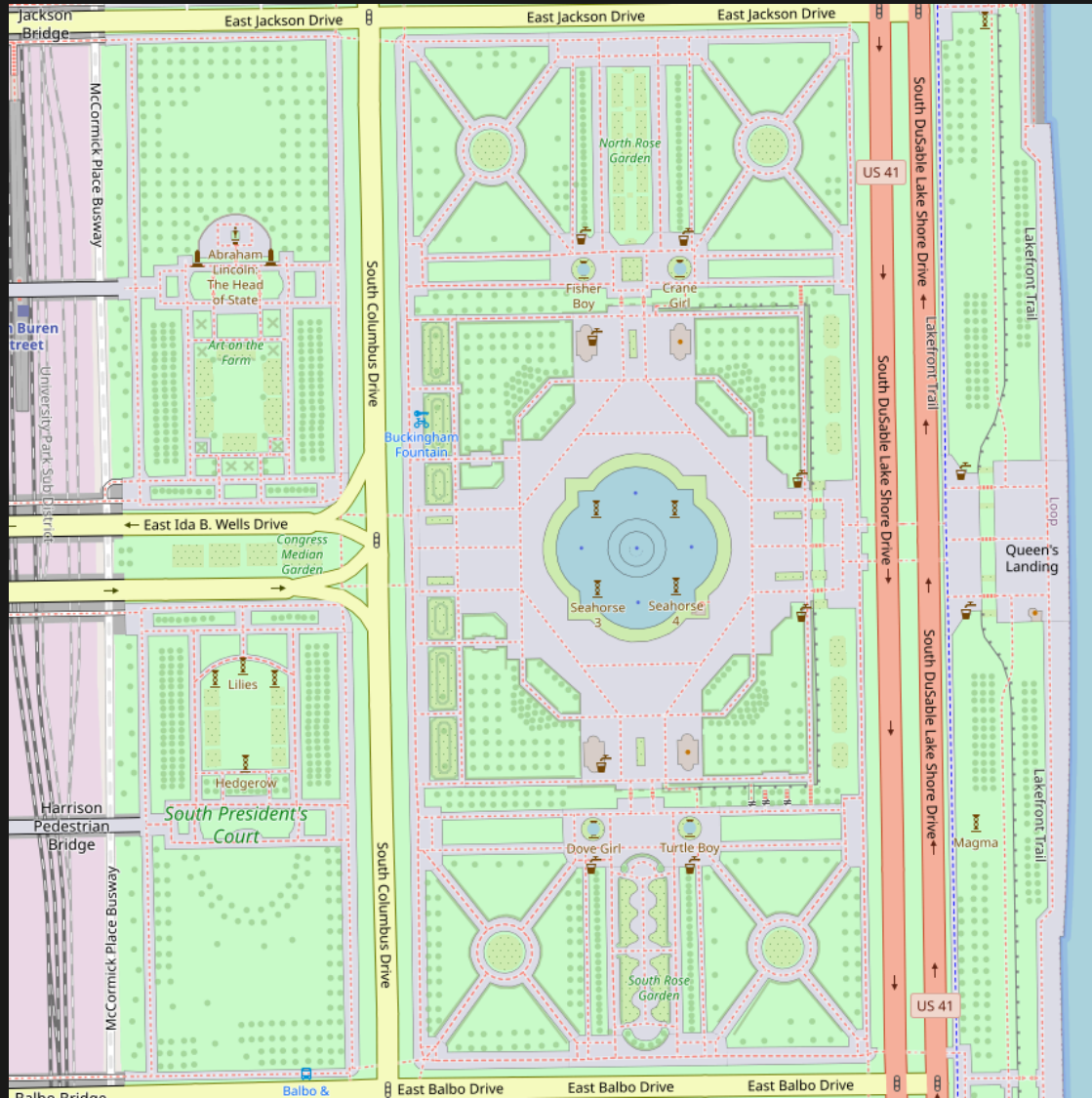


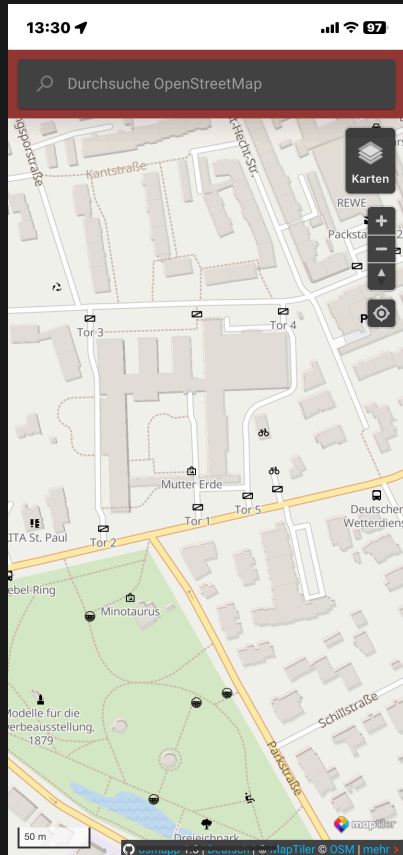
OPENSTREETMAP UND OVERPASS

OPENSTREETMAP

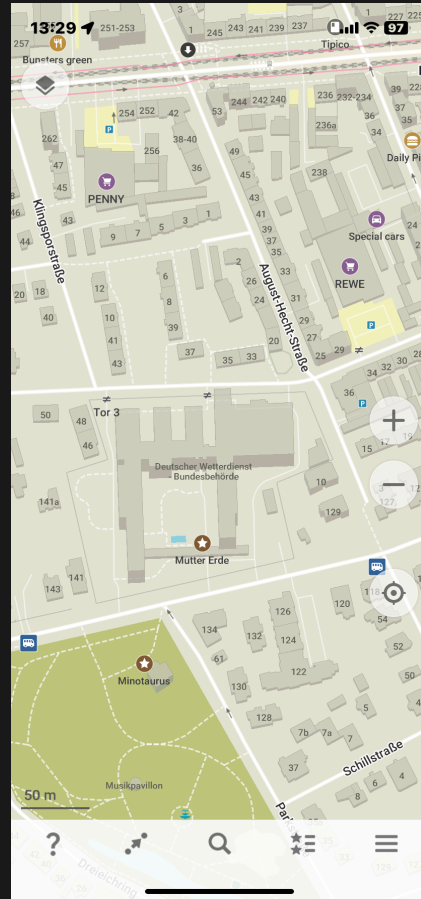
OPENSTREETMAP

- Openstreetmap ist eine freie Karte, von daher *Open data*
- Jeder kann sie bearbeiten
- Die Karte lebt von Bearbeitungen

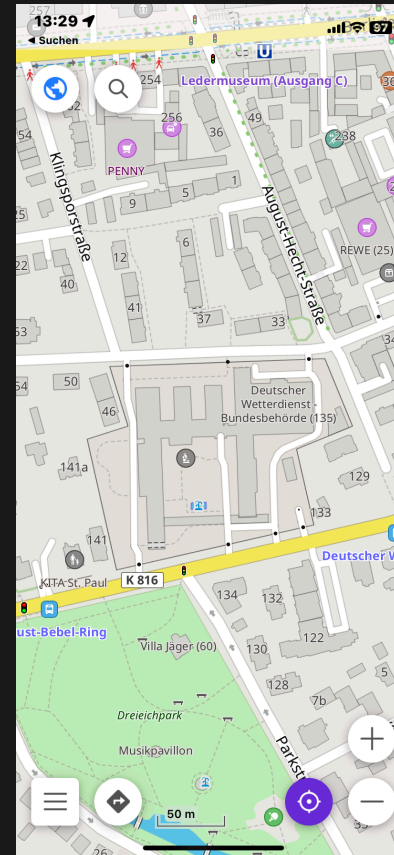




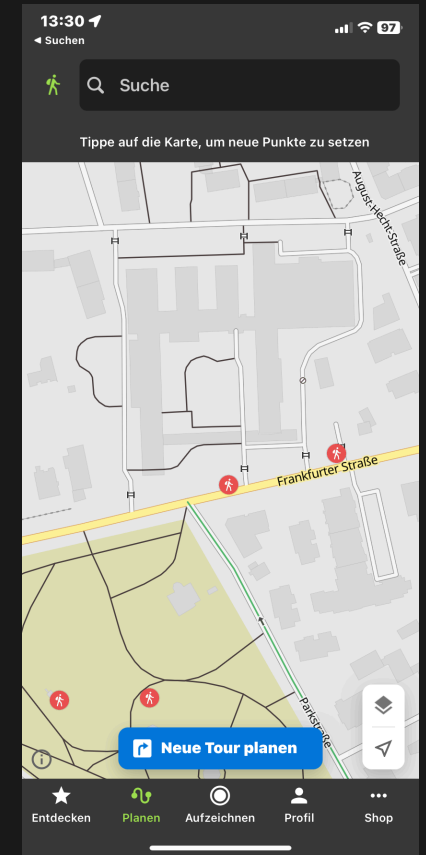
OSMAPP



ORGANICMAPS



OSMAND



KOMOOT

DATENSTRUKTUR

VERSCHIEDENE ARTEN VON OBJEKTEN

NODE

Ein Punkt auf der Karte

- Die meisten Sitzbänke
- Viele Läden

WAY

Eine Linie oder eine simpele Fläche

- Eine Straße
- Ein See
- Ein Gebäude

RELATION

Ein Zusammenschluss von Nodes, Ways und Relations

- Ein Gebäude mit *Loch*
- Eine Buslinie

FRAGERUNDE

TAGS

Tags sind Schlüssel-Wert-Paare

```
1 highway=primary
2 name=Hauptstraße
3 maxspeed=80
4 lanes=4
5 lanes:forward=3
6 lanes:backward=1
7 surface=asphalt;concrete
```

TAGS

Tags sind Schlüssel-Wert-Paare

```
1 highway=primary
2 name=Hauptstraße
3 maxspeed=80
4 lanes=4
5 lanes:forward=3
6 lanes:backward=1
7 surface=asphalt;concrete
```

TAGS

Tags sind Schlüssel-Wert-Paare

```
1 highway=primary
2 name=Hauptstraße
3 maxspeed=80
4 lanes=4
5 lanes:forward=3
6 lanes:backward=1
7 surface=asphalt;concrete
```

TAGS

Tags sind Schlüssel-Wert-Paare

```
1 highway=primary
2 name=Hauptstraße
3 maxspeed=80
4 lanes=4
5 lanes:forward=3
6 lanes:backward=1
7 surface=asphalt;concrete
```

OVERPASS



www.overpass-turbo.eu

WIZARD

▶ Ausführen

↻ Teilen

↗ Export

✂ Wizard

💾 Speichern

📁 Laden

⚙ Einstellungen

? Hilfe

overpass turbo 🚗

LIVE DEMO

BEISPIELE

shop=books in Berlin

cinema in tokiio

amenity=parking and fee=no in "Köln"

```
amenity=fast_food and name!="McDonald's" and name!="Burger King"
```

QUERY


```
1  /*
2  This has been generated by the overpass-turbo wizard.
3  The original search was:
4  "shop=books in Berlin"
5  */
6  [out:json][timeout:25];
7  // fetch area "Berlin" to search in
8  {{geocodeArea:Berlin}}->.searchArea;
9  // gather results
10 (
11     // query part for: "shop=books"
12     node["shop"="books"](area.searchArea);
13     way["shop"="books"](area.searchArea);
14     relation["shop"="books"](area.searchArea);
15 )
```

```
1 [out:json][timeout:25];
2 {{geocodeArea:Berlin}}->.searchArea;
3
4 (
5     node["shop"="books"](area.searchArea);
6     way["shop"="books"](area.searchArea);
7     relation["shop"="books"](area.searchArea);
8 );
9
10 out body;
11 >;
12 out skel qt;
```

```
1 [out:json][timeout:25];
2 {{geocodeArea:Berlin}}->.searchArea;
3
4 (
5     node["shop"="books"](area.searchArea);
6     way["shop"="books"](area.searchArea);
7     relation["shop"="books"](area.searchArea);
8 );
9
10 out body;
11 >;
12 out skel qt;
```

```
1 [out:json][timeout:25];
2 {{geocodeArea:Berlin}}->.searchArea;
3
4 (
5     node["shop"="books"](area.searchArea);
6     way["shop"="books"](area.searchArea);
7     relation["shop"="books"](area.searchArea);
8 );
9
10 out geom;
```

```
1 [out:json][timeout:25];  
2 {{geocodeArea:Berlin}}->.searchArea;  
3  
4 nwr["shop"="books"](area.searchArea);  
5  
6 out geom;
```

QUERY LANGUAGE

- Semikolon: **notwendig**.
- *node, way, relation*
- Kombinationen: *nw, nwr, nr, wr*.
- Tagsuche: *[shop=books][opening_hours]* für Buchhandlungen mit Öffnungszeiten.
- Mit runden Klammern kann man geographisch eingränzen und schauen ob etwas z.B. mitglied einer relation ist

SETS UND UNIONS

SETS UND UNIONS

```
1 node[shop=books];  
2 way[shop=books];  
3 relation[shop=books];
```


SETS UND UNIONS

```
1 (  
2   node[shop=books];  
3   way[shop=books];  
4   relation[shop=books];  
5 );
```

SETS UND UNIONS

```
1 node[shop=books] ->.bookStores;  
2  
3 .bookStores;  
4  
5 out center;
```

FRAGERUNDE

BEISPIELE

BEISPIELE

```
1 [out:csv(::id,::lat,::lon;true;"");];  
2  
3 // Das Camp Gelände mit der ID  
4 area(442428017)->.searchArea;  
5  
6 // Bäume in der searchArea  
7 node["natural"="tree"](area.searchArea);  
8 out;
```

BEISPIELE

```
1 [out:json][timeout:25];  
2 {{geocodeArea:Hamburg}}->.searchArea;  
3 nwr["amenity"="restaurant"][cuisine~"korean"](area.searchArea);  
4  
5 out center;
```

BEISPIELE

```
1 [out:csv(ref, colour; false; ';')];
2 node(205364328)->.center;
3
4 (
5     node(around.center:150)["public_transport"="stop_position"];
6     nw(around.center:150)["highway"="bus_stop"];
7     nwr(around.center:150)["amenity"="ferry_terminal"];
8 ) -> .stops;
9
10 rel(bn.stops)["route"~"bus|train|tram|subway|light_rail|ferry|monorail"];
11 out;
```

BEISPIELE

```
1 [out:csv(ref, colour; false; ';')];
2 node(205364328)->.center;
3
4 (
5     node(around.center:150)["public_transport"]="stop_position";
6     nw(around.center:150)["highway"]="bus_stop";
7     nwr(around.center:150)["amenity"]="ferry_terminal";
8 ) -> .stops;
9
10 rel(bn.stops)["route"~"bus|train|tram|subway|light_rail|ferry|monorail"];
11 out;
```


BEISPIELE

```
1 [out:csv(ref, colour; false; ';')];  
2 node(205364328)->.center;  
3  
4 (  
5     node(around.center:150)["public_transport"]="stop_position";  
6     nw(around.center:150)["highway"]="bus_stop";  
7     nwr(around.center:150)["amenity"]="ferry_terminal";  
8 ) -> .stops;  
9  
10 rel(bn.stops)["route"~"bus|train|tram|subway|light_rail|ferry|monorail"];  
11 out;
```

BEISPIELE

```
1 [out:csv(ref, colour; false; ';')];
2 node(205364328)->.center;
3
4 (
5     node(around.center:150)["public_transport"]="stop_position";
6     nw(around.center:150)["highway"]="bus_stop";
7     nwr(around.center:150)["amenity"]="ferry_terminal";
8 ) -> .stops;
9
10 rel(bn.stops)["route"~"bus|train|tram|subway|light_rail|ferry|monorail"];
11 out;
```

BEISPIELE

```
1 [out:csv(ref, colour; false; ';')];  
2 node(205364328)->.center;  
3  
4 (  
5     node(around.center:150)["public_transport"="stop_position"];  
6     nw(around.center:150)["highway"="bus_stop"];  
7     nwr(around.center:150)["amenity"="ferry_terminal"];  
8 ) -> .stops;  
9  
10 rel(bn.stops)["route"~"bus|train|tram|subway|light_rail|ferry|monorail"];  
11 out;
```

BEISPIELE

```
1 [out:csv(ref, colour; false; ';')];  
2 node(205364328)->.center;  
3  
4 (  
5     node(around.center:150)["public_transport"="stop_position"];  
6     nw(around.center:150)["highway"="bus_stop"];  
7     nwr(around.center:150)["amenity"="ferry_terminal"];  
8 ) -> .stops;  
9  
10 rel(bn.stops)["route"~"bus|train|tram|subway|light_rail|ferry|monorail"];  
11 out;
```

BEISPIELE

```
1 [out:csv(ref, colour; false; ';')];
2 node(205364328)->.center;
3
4 (
5     node(around.center:150)["public_transport"="stop_position"];
6     nw(around.center:150)["highway"="bus_stop"];
7     nwr(around.center:150)["amenity"="ferry_terminal"];
8 ) -> .stops;
9
10 rel(bn.stops)["route"~"bus|train|tram|subway|light_rail|ferry|monorail"];
11 out;
```