Kartenredaktion SoSe13 – Arbeitsblatt für TileMill

Definitionen:

Hiermit sollen nur ein paar grundlegende Elemente (von CartoCSS) erläutert werden, die benötigt werden um eine einfache Karte mit TileMill herzustellen. Es ist auch komplexer realisierbar, benötigt aber mehr Erfahrung.

Map → Hintergrund: Hiermit wird der Hintergrund der Karte definiert. Dieses Element enthält meist lediglich die Angabe der Hintergrundfarbe.

z.B.: *Map { background-color: #ddeeff }*

→ Layername : Aufruf eines importierten Datenlayers...MUSS dem, bei Import, angegebenen Namen entsprechen!

z.B.: *#countries*

{} → Definitionsbereich: Klassisches StyleSheet-Element, mit dem der Bereich vorgegeben wird innerhalb dessen Definitionen und Symbolisierungen angegeben werden.

z.B.:
```
#countries{
...
}
```

:: → „Stil“: Damit ist es möglich innerhalb der Beschreibungen eines Datenlayers einzeln definierte Symbolisierungen anzulegen...Name kann frei gewählt werden!

z.B.:
```
#countries{
    ::EU-States{
    ...
    }
    ::Non-EU-States{
    ...
    }
}
```

[] → Filter: Ermöglicht die Filterung nach unterschiedlichen Attributen. Diese können beispielsweise abhängig von Tabelleneinträgen sein (z.B. Einwohnerzahl > 1.000.000) oder sich auf den darzustellenden Maßstabsbereich beziehen (z.B. Zoom < 10)

z.B.:
```
#countries[zoom < 17][zoom > 10]{
...                    → Daten werden nur in Zoomstufen 16-11 dargestellt
}
```
oder
```
::EU-States[POP_EST > 1000000]{
    …          → Symbolisierung gilt nur wenn Wert von POP_EST größer als … ist
    }
```

!Achtung: Die Schreibweise der entsprechenden Spalte muss exakt identisch mit Datenquelle sein!

@ → Variable: Mit diesem Zeichen können Variablen deklariert werden. Das lohnt sich vor allem bei Werten die häufig genutzt werden, wie Farben oder Fonts.

z.B.: *@green: #d3e46f;*

oder *@futura_med: "Futura Medium"*

Symbolisierungen:

Die Symbolisierung wird in TileMill über sogenannte „symbolizer“ realisiert. Wenn in den Daten entsprechende Geometrien vorhanden sind werden diese nach Angabe des „symbolizers“ automatisch, mit den angegebenen Werten visualisiert!
Die wichtigsten sind:

- point (alternativ: marker)
- line
- polygon
- text

Über zusätzliche (mit Bindestrich) angefügte Bezeichnungen können dann die Parameter von Punkten, Linien, Polygonen und Text definiert werden.
Zum Beispiel:

- point-file → Angabe der Imagedatei, die angezeigt werden soll

- line-color → Linienfarbe
- line-width → Linienbreite

- polygon-fill → Farbe der Flächenfüllung
- polygon-opacity → Transparenzwert

- text-name → Angabe der Tabellenspalte oder fixer Werte die Text repräsentieren soll
- text-fill → Textfarbe

Die Syntax dieser Symbolisierungen kann auf folgende Arten geschehen:

- text-name:"[NAME]"; → Tabellenspalte
- line-color:#c0d8ff; → hexadezimaler Farbwert
- text-size:11; → Zahlenwert

Manche Symbolizer haben auch bestimmte Werte die angegeben werden müssen, die dann jeweils in der Hilfe einsehbar sind.
z.B.:

- line-cap:round;
- text-transform:uppercase;

Farben:

Hexadezimal – Kodierung: → **line-color: #ff0000** = Rot

Diese Kodierung bezieht sich auf das RGB-Farbsystem wobei jeweils 2 Buchstaben oder Zahlenwerte für eine Farbe stehen, also #RRGGBB.
Farbcodes mit 'Doppelwerten' können auch in verkürzter Schreibweise angegeben werden: → **line-color: #ff0** (steht für: #RGB)

8Bit – Farbkodierung: → **line-color: rgb(255, 255, 0)** = Rot+Grün=Gelb

Ähnliches Prinzip nur mit den Werten von 0...255 definiert.
Diese Kodierung gibt es auch noch mit Transparenz, mit Werten zwischen 0...1 (prozentuale Anagabe - 0 = transparent & 1vollfarbe)!
→ **line-color: rgba(0, 0, 255, 0.5)** = halbtransparentes Blau

Vordefinierte Farben:

Es können in TileMill auch vordefinierte Farben verwendet werden.

- TileMill-spezifische Farben → **line-color: red**
- selbst definierte Farben → **line-color: @green**