**GIS Abschlussaufgabe**

#Einwohner, KFZ Zulassungen, etc.

<http://ww2.heidelberg.de/datenatlas/>

#Konkurrenz

overpass 🡪 nodes cuisine=pizza, restaurantes  
Lieferando, Lieferheld etc. 🡪 Karte

GIS: 0-Emissionen

Voranalyse:  
  
🡪 Datenatlas: KFZ Zulassungen, Einwohnerdichte, Altersstruktur etc. = Zielgruppe

* Karte mit Lieferdiensten = Konkurrenz
* Fahrradwege & brauchbare Straßen = Netzwerk (wie viel km Fahrradweg?)
* Stadtteile 🡪 Voranalyse: welcher Stadtteil? 🡪 dann allgemeine Karte  
    
    
    
  Tools:  
  buffer, field calc, clip, intersect, dissolve, table join, Textlayer einfügen  
    
    
  Nachanalyse:
* Geeignetes Routing Plug In? 🡪 Wie schnell wie viel Haushalte? Schneller als Auto?

**Für Fahrradnetzwerk:**  
  
#This shows the cycleway and cycleroute network

*[out:json];*

*(*

*// get cycle route relatoins*

*relation[route=bicycle]({{bbox}});*

*// get cycleways*

*way[highway=cycleway]({{bbox}});*

*way[highway=path][bicycle=designated]({{bbox}});*

*);*

*out body;*

*>;*

*out skel qt;*

**Konkurrenz** amentiy=fast\_food

Amenity=restaurant  
 amenity=pizza

QGIS Analyse

**Table Join:**

* .csv Dateien von Datenatlas als *Deimited Textlayer* hinzufügen (ohne Geometrie)
* Unter Eigenschaften von Vektorlayer „Stadtteile“ auf Joins
* Hinzufügen 🡪 „kfz-dichte“ : Join Field „Name“ und Target Field „Stadtteil“ (nur bestimmte Spalten)
* Layer: stadtteile\_joined

**Union (Konkurrenz)**

* Union 🡪 amenity=pizza & amenity=fastfood
* Union 🡪 Restaurantes & italian
* Beide zusammen 🡪 Layer: konkurrenz\_gesamt

**Wie viel Konkurrenz in welchem Stadtteil?**

* Vector analysis 🡪 count points in polygon
* Polygon Layer: stadtteile\_joined
* Point Layer: Konkurrenz\_gesamt
* Ausgabelayer: stadteile\_joined2

**Merge (Straßennetzwerk)**

* Input: cycleway und residential streets
* Layer: cycle+residential

**Sum Line lengths (Berechnung wie lang linestring in einzelen Polygonen)**

* Input: cycle+residential & stadtteile\_joined2
* Ausgabelayer: stadtteile\_joined3

**2. Analyseschritt**

Netzwerkanschluss der Gebäude

1) Intersection🡪 Handschuhsheim+Neuenheim & Cyclenetwork

2) Buffer🡪 Buffer 10m

3) Extract by location 🡪 extract from: buildings H+N; comparing with 10mBuffer\_cycle (intersect, touch, overlap, within, cross, contain)

🡪 **Ausgabelayer = buildings\_streetnetwork**