

Schriftliche Aufgabenbearbeitung und Dokumentation

Aufgabe 1

Schaue dir die Datensätze hochschulen.csv1 und bevoelkerung.csv2 an.

Konzipiere eine Datenvisualisierung, die Informationen aus den gegebenen Datensätzen integriert und effektiv präsentiert. Bereite eine kurze schriftliche Beschreibung deines Projektziels auf.

Konzept der Datenvisualisierung:

- Darstellung als Bubbles auf einer Karte
- Anzahl Studierende pro Ort
- Anzahl Studierende pro Bundesland in Relation zur Bevölkerungsmenge
- Anzahl öffentlich-rechtliche
- Weibliche/Männliche Bevölkerung in einem bestimmten Bundesland

Projektziel:

Die Ergebnisse sind auf einer Karte dargestellt, sodass die Werte leicht vergleichbar sind.

Abänderung des Ziels:

Wir haben keine Möglichkeit gefunden und zum Laufen bekommen, mit Python und Bibliotheken eine Datei mit den Koordinaten der Orte zu generieren und per Hand wäre es für die Menge der Orte zu zeitaufwändig gewesen. Deshalb haben wir uns dazu entschieden, die Daten aus den Abfragen doch in Form von Balkendiagrammen darzustellen.

Aufgabe 2

1. Identifiziere die relevanten Datenfelder für die Analyse und Visualisierung.

Datenbereinigung:

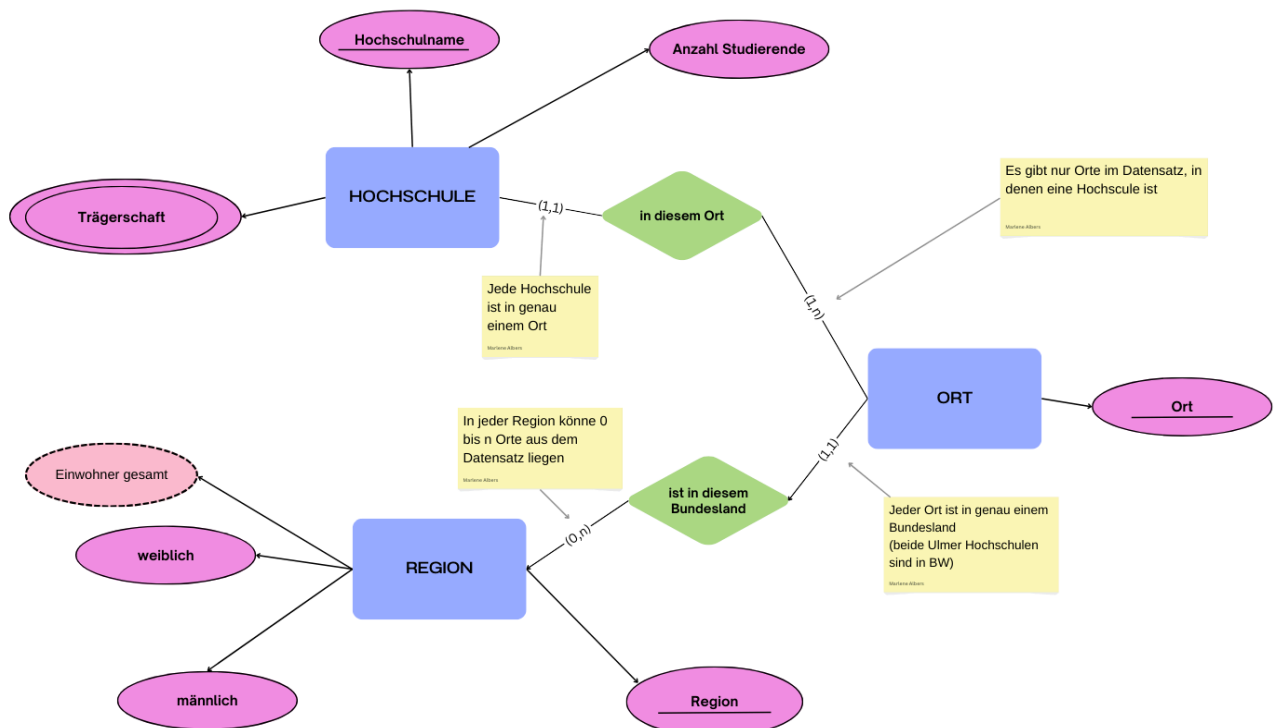
rausnehmen:

- Homepage (Brauchen wir nicht für unsere Abfragen und hilft uns nicht auf einen anderen Datensatz zuzugreifen)
- Hochschulkurzname (Wir haben den ganzen Hochschulnamen schon)
- Hochschultyp (Brauchen wir nicht für unsere Abfragen und hilft uns nicht auf einen anderen Datensatz zuzugreifen)
- Gründungsjahr (Brauchen wir nicht für unsere Abfragen und auch nicht um auf den anderen Datensatz zuzugreifen)
- Habilitationsrecht (Brauchen wir nicht für unsere Abfragen und auch nicht um auf den anderen Datensatz zuzugreifen)
- Promotionsrecht (Brauchen wir nicht für unsere Abfragen und hilft uns nicht auf einen anderen Datensatz zuzugreifen)
- Straße (Brauchen wir nicht für unsere Abfragen und auch nicht um auf den anderen Datensatz zuzugreifen)
- Bundesland (Nehmen wir aus HOCHSCHULE raus und packen es in eine neue Tabelle ORT, damit wir damit später von ORT auf REGION kommen)

- DG (Brauchen wir nicht für Abfrage und brauchen wir auch nicht um auf anderen Datensatz zuzugreifen, da wir uns nur auf die Bundesländer fokussieren um von der einen Datenbank auf die andere zu kommen)

2. Zeichne ein Entity-Relationship-Modell (ERM) für die Datensätze.

Entity Relationship Model



3. Transformieren Sie das ERM in ein relationales Modell.

- HOCHSCHULE (Hochschulname, Trägerschaft, Anzahl_Studierende, Ort)
- ORT(Ort, Region)
- REGION (Region, weiblich, männlich)

Fremdschlüssel (keine gestrichelte Linie gefunden)

Dokumentation: Wir haben drei Tabellen, von Hochschule kann man Ort über das Attribut Ort (in HOCHSCHULE) und Ortsname (in ORT) erreichen und von Ort kommt man zu Region über Bundesland (in ORT) und Region (in REGION) erreichen.

4. Richte eine PostgreSQL-Datenbank basierend auf Ihrem relationalen Modell ein
Die Codes dafür sind in der ZIP

Aufgabe 3 Datenbereinigung

- Zeilen mit fehlenden Werten (in der Tabelle steht da ein Minus) löschen (Nur wenn Daten fehlen, die wir für unsere Abfrage brauchen, denn wenn diese fehlen, dann bringen uns die Zeilen nichts und dann müssen sie auch nicht in der Datenbank sein)
- Ä, Ö, Ü wurden zu ua, uo und ue gemacht, da wir sonst immer eine Fehlermeldung bekommen haben.
- Vor allen Bundesländern in der Tabelle REGION wurden die Leerzeichen gelöscht, damit diese gleich den Bundesländern in der ORT Tabelle sind.

Aufgabe 4 Queries and Analysis

1. Formuliere spezifische SQL-Abfragen, um relevante Informationen aus den Daten zu extrahieren. Beispiele: Anzahl der Studierenden pro Stadt, Geschlechterverteilung der Studierenden in den Bundesländern, Vergleich der Studierendenzahlen und Bevölkerungszahlen in verschiedenen Regionen.

Queries sind in der ZIP Datei

- Backnang, Nienburg und Bad Eilsen haben keinen Wert in Anzahl_Studierende und sind deshalb direkt unter den höchsten Werten
- ß nicht in Graph darstellbar

Aufgabe 5 Visualisierung:

- Wir benutzen GeoPanda, um die Ergebnisse unserer Abfragen darzustellen.
- Codes sind extra in der Zip Datei und sind Kommentiert
- Graphiken sind auch extra in der Zip Datei