1.Übung

# Aufgabe 1)

Beim Importieren der Datenbank bin ich wie folgt vorgegangen. Zuerst habe ich psql installiert. Dort habe ich dem User „postgres“ ein Passwort gegeben und eine neue database erstellt. Weil die in der Aufgabe verlinkte Anleitung nicht mehr aktuell war, habe ich durch viel ausprobieren versucht es selbst hinzubekommen. Nach einigen Versuchen bin ich dann auf die Idee gekommen bei den Issues zu schauen. Dort hat jemand sein Skript hochgeladen, mit dem die Daten importieren kann. Es war aber doch nicht so einfach wie gedacht. Als ich das Skript ausführen wollte durfte ich das nicht, weil die CSV-Files, die dort verwendet wurden zum Importieren „WIN1512“ Encoding verwenden und meine Datenbank nicht. Also musste ich durch viel herumprobieren das Encoding meiner DB ändern und konnte danach auch die Daten importieren. Die daraus resultierenden Daten habe ich dann nochmal als CSV-File exportiert um für jeden Table, ein CSV-File zu bekommen. Diese sind auch meine Daten, die ich für die Lösung der nächsten Aufgabe verwende.

# Aufgabe 2)

Ich dachte ursprünglich, dass wir die einzelnen queries die wir uns ausgesucht haben, mithilfe von pandas implementieren sollen. Also habe ich für die erste Query, die Abhängigkeiten aufgelöst, und selbst versucht zu implementieren. Das ist mir auch gelungen, war aber extrem zeitaufwändig. Dann habe ich nach alternativen Packages geschaut, die mir diese Aufgabe abnehmen können und bin auch fündig geworden. Zuerst habe ich mit „pandasql“ versucht die CSV-Files zu durchsuchen, aber musste feststellen das die Laufzeit von dem Package bei den Queries viel zu hoch ist. Nach weiterer Recherche habe ich dann „duckdb“ entdeckt. Ein sehr einfach zu nutzendes Package, welches CSV-Files direkt mittels sql-queries durchsuchen kann. Damit konnte ich in kurzer Zeit die CSV-Files durchsuchen.  
Anfangs habe ich versucht mit meinem Laptop die Aufgabe zu lösen, doch leider habe ich zu wenig RAM um diese Aufgabe ohne großen Mehraufwand bewältigen. Anfangs habe ich durch Chunks versucht die großen Daten sequenziell zu durchsuchen, habe mich aber dann doch dafür entschieden meinen Rechner zu verwenden. Dort war das dann kein Problem mehr, aber ich musste die DB nochmal neu aufsetzten.

# Aufgabe 3)

Das Verbinden zur PSQL-DB war sehr einfach durch Psycopg2 zu implementieren. Der Cursor muss nur mit der entsprechenden query gefüttert werden und liefert das erwartete Ergebnis. Dann musste ich nur noch die Lösung aus Aufgabe 2 in ein vergleichbares Format bringen, um zu prüfen ob die Ergebnisse gleich sind.

Verwendeter Code: <https://github.com/DerTuchi/dbi>