

Title - Assessment Project for Working with Databases using Pandas and Python.

GitHub Repo - <https://github.com/bleso-a/database-with-pandas-and-python>

Part 1 - Query languages in Nigeria from “World_x” MySQL Database

The aim of this assessment is to check your skills of communicating with a MySQL relational database management system from the Python environment. You are expected to demonstrate mastery of your skills to create connections, use Pandas to read tables & run queries.

Performing Queries using Pandas.

In this project you are expected to work with the MySQL sample database “world_x” database
Link- <https://dev.mysql.com/doc/index-other.html>

File - Starter_code_part1

Step 1 - Create an SQLAlchemy engine for a MySQL database, and enter the appropriate credentials, which includes user, password, database and host.

Step 2 - Read the “country” and the “countrylanguage” tables. You can do this with the Pandas.read_sql or read_sql_table method.

You expected to generate two pandas DataFrame from this, and you can provide any name for them.

Step 3 - Rename the “CountryCode” column in the DataFrame generated from the “Countrylanguage” table. The new column name is “Code”. Hint: Use the Pandas.rename() method for this.

Step 4 - In this step, you are expected to perform a LEFT JOIN by joining the “country” table to “countrylanguage”. You will perform this step using the Pandas method of combining data on a common column which is “Code”. Save your result to a DataFrame named “df_country”

Step 5 - From the results above, you will filter the DataFrame(df_country) to return another DataFrame where the “Code” column is “NGA” and the “Name” column is “Nigeria”. In essence, you are to generate a table for the country Nigeria. You can name this DataFrame “df_Nigeria”

Step 6 - To test the methods above, perform a similar LEFT JOIN by writing an SQL query(not Pandas method) and save to a variable. Using the Pandas.read_sql method, read the query

from the database to perform and generate results for the join procedure. Save the result to another DataFrame name e.g df_Nigeria_SQL

Note: Use SELECT * when writing the query so as to generate columns that are similar to previous steps. Also remember to drop the “countrycode” column

Step 7 - Use the shape attribute to check the shape of both DataFrames(df_Nigeria & df_Nigeria_SQL). The result should be the same for both DataFrames. You can write a conditional statement to check if both shapes are equal.

Ensure that you don’t have “countrycode” column in your test DataFrame

Step 8 - Using the DataFrame.to_sql method, save the results of “df_Nigeria” to a new table(you can provide any name e.g “nigerianlanguage” to the database.

Provide a screenshot to show that your procedure ran successfully.

Part 2 - Fetching customer records from DVD rental PostgreSQL Sample Database

The aim of this assessment is to check your skills of working with a PostgreSQL relational database management system from the Python environment. You are expected to demonstrate mastery of your skills to create connections, and perform core database management tasks. You will write functions to fetch customer records from the sample database.

In this assessment, you will work with the PostgreSQL sample database. Please refer to the course for a recap on how to load data to the database.

Link - <https://www.postgresqltutorial.com/postgresql-sample-database/>

File - starter_code_part2

You have a file starter_code.py which contains starter codes, you are expected to fill the spaces where we have “Write Your Code Here”.

Step 1: Complete the function for creating connection to the database. Input your connection credentials.

Step 2 - The function to close the connection has been created.

Step 3 - You have a function to read customer records from the table (getCustomer). In the function, using a try/except block, you will execute the query to read a particular customer using the “customer_id” as the getCustomer parameter.

Follow the “Write Your Code” instructions to complete this step.

Step 4 - Take a screenshot of your results in the terminal, and include in your submission.

