**Assignment**

**CSA0805 – Python Programming**

|  |  |
| --- | --- |
| **Register Number** | **192324254** |
| **Name** | **S. SASI KUMAR** |

**Title:**

**Database Query Tool Using SQL Alchemy**

**Problem Statement:**

**The objective is to develop a Python program that connects to a database using the SQL Alchemy module. The program should allow users to execute SQL queries either interactively or from script files and provide the ability to view or export the query results.**

**Code:**

**from sqlalchemy import create\_engine, text**

**import pandas as pd**

**class DatabaseQueryTool:**

**def \_init\_(self, db\_url):**

**# Connect to the database**

**self.engine = create\_engine(db\_url)**

**def execute\_query(self, query):**

**# Execute a single query**

**with self.engine.connect() as connection:**

**result = connection.execute(text(query))**

**# Fetch all rows**

**return result.fetchall()**

**def execute\_script(self, script\_file):**

**# Execute queries from a script file**

**with open(script\_file, 'r') as file:**

**script = file.read()**

**with self.engine.connect() as connection:**

**connection.execute(text(script))**

**print(f"Executed script from {script\_file}")**

**def export\_query\_results(self, query, output\_file):**

**# Execute query and export results to a CSV file**

**df = pd.read\_sql\_query(text(query), self.engine)**

**df.to\_csv(output\_file, index=False)**

**print(f"Results exported to {output\_file}")**

**def interactive\_mode(self):**

**# Allow users to execute queries interactively**

**print("Interactive SQL Query Mode. Type 'exit' to quit.")**

**while True:**

**query = input("SQL> ")**

**if query.lower() == 'exit':**

**break**

**try:**

**results = self.execute\_query(query)**

**for row in results:**

**print(row)**

**except Exception as e:**

**print(f"Error: {e}")**

**# Replace with your actual database connection string**

**db\_url = 'mysql+pymysql://username:password@localhost/dbname'**

**tool = DatabaseQueryTool(db\_url)**

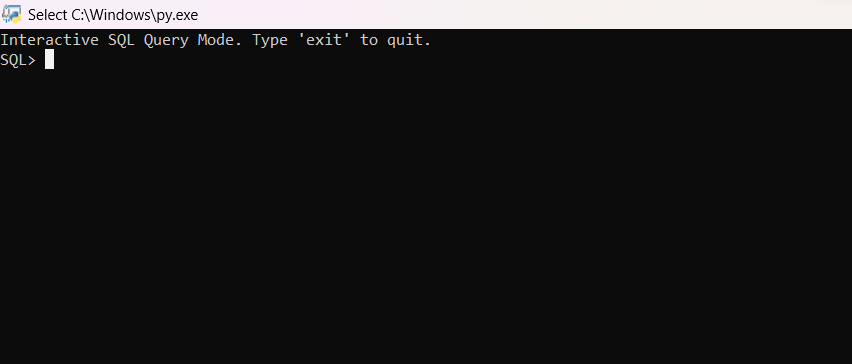
**# Example usage**

**tool.interactive\_mode() # Start the interactive mode**

**# tool.execute\_script('script.sql') # Execute SQL from a file**

**# tool.export\_query\_results('SELECT \* FROM table\_name', 'output.csv') # Export query results to a CSV**

**Output Screen Shots:**

****

**Conclusion:**

**The Python program successfully connects to a database using SQLAlchemy, allowing users to execute SQL queries interactively or from script files. The program also provides the capability to export query results to CSV files. This tool is versatile and can be extended to support different databases by simply changing the connection string.**