IMPORT AND SETUP PACKAGES

```
# pymysql, sqlalchemy, pandas, python-dotenv
! pip install pymysql sqlalchemy pandas python-dotenv
```

```
Collecting pymysql
  Downloading pymysql-1.1.2-py3-none-any.whl.metadata (4.3 kB)
Requirement already satisfied: sqlalchemy in /usr/local/lib/python3.12/dist-packages (2.0.44)
Requirement already satisfied: pandas in /usr/local/lib/python3.12/dist-packages (2.2.2)
Requirement already satisfied: python-dotenv in /usr/local/lib/python3.12/dist-packages (1.1.1)
Requirement already satisfied: greenlet>=1 in /usr/local/lib/python3.12/dist-packages (from sqlalchemy) (3.2.4)
Requirement already satisfied: typing-extensions>=4.6.0 in /usr/local/lib/python3.12/dist-packages (from sqlalc
Requirement already satisfied: numpy>=1.26.0 in /usr/local/lib/python3.12/dist-packages (from pandas) (2.0.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.12/dist-packages (from pandas)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.12/dist-packages (from pandas) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.12/dist-packages (from pandas) (2025.2
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.12/dist-packages (from python-dateutil>=2.8.2
Downloading pymysql-1.1.2-py3-none-any.whl (45 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 45.3/45.3 kB 3.1 MB/s eta 0:00:00
Installing collected packages: pymysql
Successfully installed pymysql-1.1.2
```

SETUP TO CREATE ENVIRONMENT + CHECK

```
from sqlalchemy import create_engine
import pandas as pd

import os
from dotenv import load_dotenv

load_dotenv('.env')
```

```
True
```

CONNECT TO DATABASE USING .ENV

```
# Set up connection string from .env variables

sql_username = os.getenv('username')
sql_password = os.getenv('password')
sql_host = os.getenv('hostname')
sql_port = os.getenv('port')
sql_database = os.getenv('database')
```

TEST CONNECTION

```
sql_username
```

```
'ahistudent'
```

```
sql_port
```

```
'3306'
```

CREATE URL CONNECTION

```
# w.o SSL on

url_string = f"mysql+pymysql://{sql_username}:{sql_password}@{sql_host}:{sql_port}/{sql_database}"
```

```
conn = create_engine(url_string)
```

SQL QUERY: PREVIEW FIRST 50 ROWS

```
# query first 50 rwos from table

sql_toexecute = """
  select *
  from research_experiment_refactor_test
  limit 50;
"""
```

```
# display first 50 rwos from the table

response = pd.read_sql(sql_toexecute, conn)
response
```

Show hidden output

SETUP ENABLING INTERACTIVE TABLE VIEW

```
from google.colab import data_table
```

```
data_table.enable_dataframe_formatter()

```

CREATING CONNECTION ENGINE TO DATABASE

```
# Create connection string for MySQL using PyMySQL
conn_str = f"mysql+pymysql://{sql_username}:{sql_password}@{sql_host}:{sql_port}/{sql_database}"
engine = create_engine(conn_str)
```

ADDING KEY SQLALCHEMY COMPONENTS TO INTERACT WITH THE RELATIONAL DATABASE

```
from sqlalchemy import create_engine, MetaData, Table, select

```

EXPLORING BASIC DATA

ROW COUNT

```
# Query to get the total rows count from the table
count_query = 'SELECT COUNT(*) AS total_rows FROM research_experiment_refactor_test'

# Execute query and get result
count_df = pd.read_sql(count_query, engine)
total_rows = count_df['total_rows'][0]
print(f'Total rows scanned from research_experiment_refactor_test: {total_rows}')
```

```
Total rows scanned from research_experiment_refactor_test: 6617426
```

METADATA: COLUMNS + DATATYPES

```
# Reflect metadata for the single known table
metadata = MetaData()
table_name = 'research_experiment_refactor_test'

# Load table schema
table = Table(table_name, metadata, autoload_with=engine)

```

```
'
# 1. Explore columns and types in the table
print(f"Columns and data types for table '{table_name}':")
cols = [(col.name, col.type) for col in table.columns]
cols_df = pd.DataFrame(cols, columns=['Column Name', 'Data Type'])
display(cols_df)
```

Columns and data types for table 'research_experiment_refactor_test':

1 to 12 of 12 entries Filter

index	Column Name	Data Type
0	id	BIGINT
1	playername	VARCHAR(255) COLLATE "utf8mb4_unicode_ci"
2	timestamp	DATETIME
3	device	VARCHAR(50) COLLATE "utf8mb4_unicode_ci"
4	metric	VARCHAR(255) COLLATE "utf8mb4_unicode_ci"
5	value	DECIMAL(20, 6)
6	team	VARCHAR(255) COLLATE "utf8mb4_unicode_ci"
7	session_type	VARCHAR(255) COLLATE "utf8mb4_unicode_ci"
8	session_description	TEXT COLLATE "utf8mb4_unicode_ci"
9	function_description	VARCHAR(255) COLLATE "utf8mb4_unicode_ci"
10	data_source	VARCHAR(50) COLLATE "utf8mb4_unicode_ci"
11	created_at	TIMESTAMP

Show 25 per page

SAMPLE DATA

```
# 2. Query sample data from the table (limit 20)
query = select(table).limit(20)
with engine.connect() as conn:
    result = conn.execute(query)
    sample_df = pd.DataFrame(result.fetchall(), columns=result.keys())
print(f"Sample data from '{table_name}':")
display(sample_df)
```

Sample data from 'research_experiment_refactor_test':

1 to 20 of 20 entries Filter

index	id	playername	timestamp	device	metric	value	team	session
0	1	PLAYER_1175	2018-10-15 19:27:41	hawkins	Avg. Braking Force(N)	"1546.381600"	Team: Stony Brook	
1	2	PLAYER_1175	2018-10-15 19:27:41	hawkins	Avg. Braking Power(W)	"-1099.217500"	Team: Stony Brook	
2	3	PLAYER_1175	2018-10-15	hawkins	Avg. Braking	"-0.763000"	Team:	

			19:27:41		Velocity(m/s)		Stony Brook	
3	4	PLAYER_1175	2018-10-15 19:27:41	hawkins	Avg. Landing Force(N)	"1251.360000"	Team: Stony Brook	
4	5	PLAYER_1175	2018-10-15 19:27:41	hawkins	Avg. Propulsive Force(N)	"1887.042100"	Team: Stony Brook	
5	6	PLAYER_1175	2018-10-15 19:27:41	hawkins	Avg. Propulsive Power(W)	"2667.976900"	Team: Stony Brook	
6	7	PLAYER_1175	2018-10-15 19:27:41	hawkins	Avg. Propulsive Velocity(m/s)	"1.458500"	Team: Stony Brook	
7	8	PLAYER_1175	2018-10-15 19:27:41	hawkins	Avg. Relative Propulsive Force(%)	"197.677300"	Team: Stony Brook	
8	9	PLAYER_1175	2018-10-15 19:27:41	hawkins	Avg. Relative Propulsive Power(W/kg)	"27.417400"	Team: Stony Brook	
9	10	PLAYER_1175	2018-10-15 19:27:41	hawkins	Braking Net Impulse(N.s)	"122.075800"	Team: Stony Brook	
10	11	PLAYER_1175	2018-10-15 19:27:41	hawkins	Braking Phase %	"25.618800"	Team: Stony Brook	
11	12	PLAYER_1175	2018-10-15 19:27:41	hawkins	Braking Phase(s)	"0.207000"	Team: Stony Brook	
12	13	PLAYER_1175	2018-10-15 19:27:41	hawkins	Braking RFD(N/s)	"4111.111100"	Team: Stony Brook	
13	14	PLAYER_1175	2018-10-15 19:27:41	hawkins	Countermovement Depth(m)	"-0.322900"	Team: Stony Brook	
14	15	PLAYER_1175	2018-10-15 19:27:41	hawkins	Flight Time(s)	"0.585000"	Team: Stony Brook	
15	16	PLAYER_1175	2018-10-15 19:27:41	hawkins	Impulse Ratio	"2.177100"	Team: Stony Brook	
16	17	PLAYER_1175	2018-10-15 19:27:41	hawkins	Jump Height(m)	"0.379500"	Team: Stony Brook	
17	18	PLAYER_1175	2018-10-15 19:27:41	hawkins	Landing Stiffness(N/m)	"-1299.216800"	Team: Stony Brook	
18	19	PLAYER_1175	2018-10-15 19:27:41	hawkins	L|R Avg. Braking Force(%)	"1.546700"	Team: Stony Brook	
19	20	PLAYER_1175	2018-10-15 19:27:41	hawkins	L|R Avg. Braking RFD(%)	"21.739100"	Team: Stony Brook	