SQL

More SQL Cheat Sheets: here
Libraries Used:
Import mysql.connector

- Used to connect Python to SQL database

Functions Used in Python:
mysql.connector.connect (more)

- Creates connection to MySQL server
- Parameters are user ('wsa'), password ('LeBron>MJ!'), host ('34.68.250.121'), and database ('Tutorials-Fall2024')

```
cnx = mysql.connector.connect(user = "wsa",
                              host = "34.68.250.121",
                              database = "Tutorials-Fall2023",
                              password = "LeBron>MJ!")
```

.cursor() (more)

- Creates a cursor object in Python that is used to execute SQL commands
- One parameter (buffered = True)

SQL statement

- Basic statements that SQL can comprehend and subsequently execute actions that we want
- In Python, we use this to insert our data into our SQL table

.execute() (more)

- Used to execute SQL statements
- We use it to execute the statement that will insert our data into the SQL table

.commit() (more)

- Verifies and completes all of the executions made by a cursor object

Navigating SQL and Creating Tables:
Creating Tables

- Schemas > Tutorials-Fall2024 > Tables (right click) > Create Table
- To update table attributes (columns & data types), hover over specified table and click on the wrench
- To view contents of table, hover over table and click on the lightning bolt

Data types (more)

- SQL can use a variety of data types, but we are only interested in a select few
- INT is used to store integer values
- VARCHAR(45) is used to store string values
  - 45 character limit

Table Constraints (more)

- When creating each column in the table, there are 8 constraints you can activate
- Primary Key (PK) uniquely identifies each row
- Not Null (NN) ensures a value cannot be null
- Auto Increment (AI) increases the value by 1 with each next row
- Unique (UQ) ensures each value is different
- Binary (B) means only binary values can be in the column (0s or 1s)
- The rest don't have importance to us

Default value

- If there is no value passed into the table, SQL will automatically fill the slot with a default value
- Typically, we use NULL for these values

SQL Statements:
SELECT FROM (more)

- Allows us to select certain values from our table
- SELECT * FROM gives us all columns in each row
  - Format: SELECT [what you want] FROM [SQL schema].[specific table];
- We can select specific columns by replacing * with the column names, separated by commas

```
SELECT playerName, weight FROM `Tutorials-Winter2023`.NBA_rosters_justin_yang
```

WHERE clause (more)

- Allows us to only select certain data where a specific clause is met
- WHERE … added to end of SELECT statement

```
SELECT playerName, weight FROM `Tutorials-Winter2023`.NBA_rosters_justin_yang
WHERE weight >= 220;
```

ORDER BY (more)

- Allows us to sort a table by values
- Automatically sorts from low value to high but adding DESC at the end switches the outcome

```
1  SELECT * FROM `Tutorials-Winter2023`.NBA_rosters_justin_yang
2  ORDER BY experience DESC;
```

Result Grid | Filter Rows: | Edit:

id	playerName	position	height	weight	experience
6	Andre Iguodala	SF	6-6	215	17
3	Stephen Curry	PG	6-2	185	12
5	Draymond Green	PF	6-6	230	9
13	Otto Porter Jr.	PF	6-8	198	8
14	Klay Thompson	SG	6-6	215	8
17	Andrew Wiggins	SF	6-7	197	7

INSERT INTO (more)

- Allows us to insert data into the SQL table
- Format: INSERT INTO [table name] ([column names]) VALUES ([values]);

```
1  INSERT INTO `Tutorials-Winter2023`.NBA_rosters_justin_yang
2  (playerName, position, height, weight) VALUES ('Billy Bob', 'C', '6-5', 300);
```

UPDATE (more)

- Allows us to update 1 or more values in the table
- Helpful when cleaning our data
- Format: UPDATE [table name] SET [variable = something] WHERE [condition is met];

```
4  UPDATE `Tutorials-Winter2023`.NBA_rosters_justin_yang SET playerName = 'Joe Schmo' WHERE id = 18;
```

DELETE (more)

- Allows us to delete information from our tables
- Format: DELETE FROM [table name] WHERE [condition is met]

```
5  DELETE FROM `Tutorials-Winter2023`.NBA_rosters_justin_yang WHERE id = 18;
```