Assignment:-03

Introduction to sql and sql commands

Instructions:

Please share your answers filled in line in the Word document. Submit code separately wherever applicable.

Please ensure you update all the details:

Name: ULLI VENKATA SAI KUMAR Batch ID: 04072024HYD10AM

Topic: Introduction to sql and sql commands

What is SQL, and what are some common uses for it in database management?

Structured query language (SQL) is a programming language for storing and processing information in a relational database. A relational database stores information in tabular form, with rows and columns representing different data attributes and the various relationships between the data values. You can use SQL statements to store, update, remove, search, and retrieve information from the database. You can also use SQL to maintain and optimize database performance.

What is a foreign key in SQL, and how is it used to establish relationships between tables?

A **foreign key** in SQL is a column or a set of columns in one table that is used to establish a link between the data in two tables. It acts as a reference to the primary key in another table, thereby creating a relationship between the two tables.

Key Concepts:

Primary Key: A column (or set of columns) in a table that uniquely identifies each row in that table. Each table can have only one primary key.

Foreign Key: A column (or set of columns) in one table that refers to the primary key in another table. The table with the foreign key is often called the "child" table, and the table with the referenced primary key is the "parent" table.

How Foreign Keys Establish Relationships:

Referential Integrity: Foreign keys help maintain referential integrity in the database. This means that a foreign key ensures that the value in the child table corresponds to a valid, existing value in the parent table. For example, if StudentID is a foreign key in the Grades table that references the StudentID primary key in the Students table, then every StudentID value in Grades must exist in Students.

One-to-Many Relationship: The most common relationship established by foreign keys is a one-to-many relationship. For instance, one student (in the Students table) can have many grades (in the Grades table), but each grade must correspond to one specific student.

Assignment:-03

Introduction to sql and sql commands

Cascading Actions: Foreign keys can be set up with cascading actions such as ON DELETE CASCADE or ON UPDATE CASCADE. These actions specify what happens in the child table when the referenced primary key in the parent table is deleted or updated. For example, ON DELETE CASCADE would automatically delete all related rows in the child table when a row in the parent table is deleted.

DATABASE CREATE:-

1. Create a database ‘classroom’

   ```
   CREATE DATABASE CLASSROOM;
   ```

2. Create a table named ‘science_class’ with the following properties

   3 columns(enrollment_no int, name varchar, science_marks int)

   ```
   USE CLASSROOM;
   ```

   ```
   CREATE TABLE SCIENCE_CLASS(ENROLLMENT_NO INT, NAME VARCHAR(20), SCIENCE_MARKS INT);
   ```

INSERTING & IMPORTING:-

1. Insert the following data into science_class using insert into command

1	popeye	33
2	olive	54
3	brutus	98

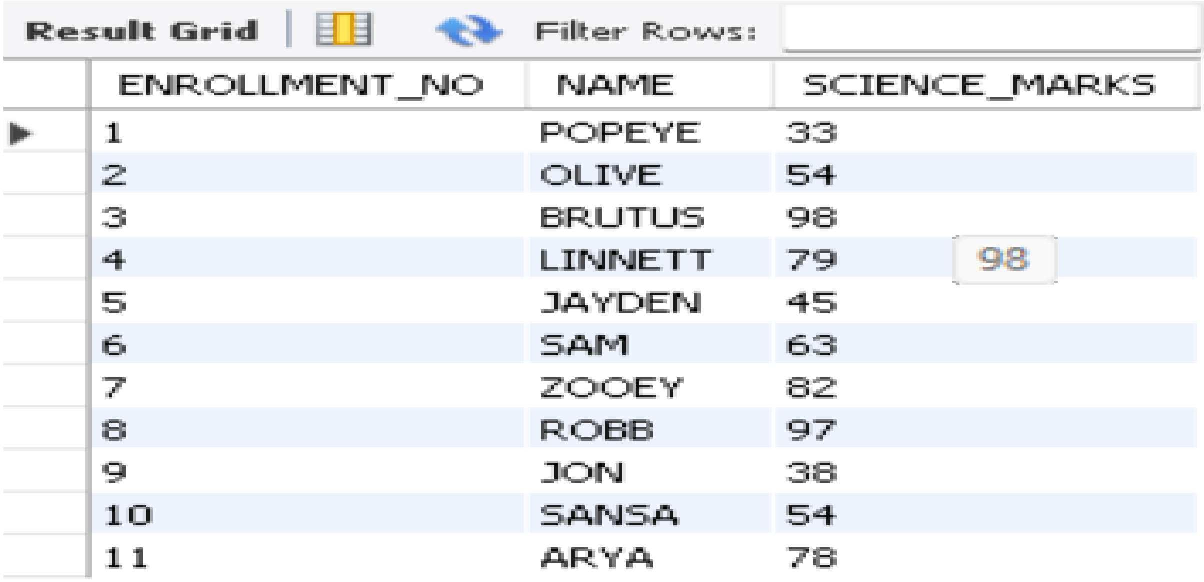
2. Import data from CSV file ‘student.csv’ attached in resources to science_class to insert data of next 8 students

360 DigiTMG®

Digital Transformation | Management | Governance

Assignment:-03

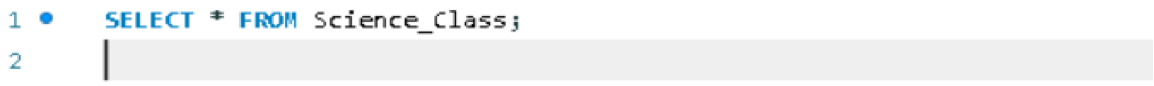
Introduction to sql and sql commands



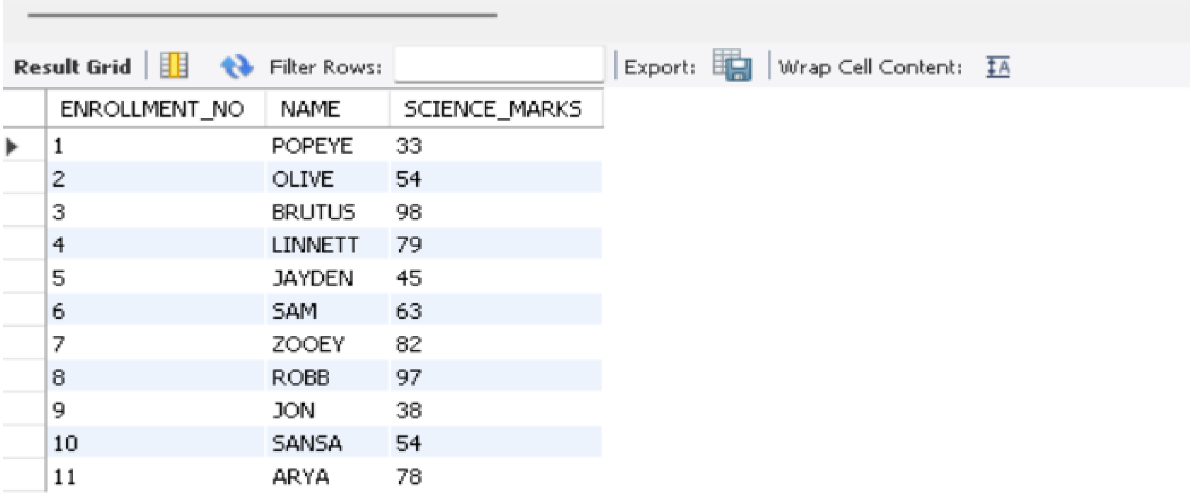
ENROLLMENT_NO	NAME	SCIENCE_MARKS
1	POPEYE	33
2	OLIVE	54
3	BRUTUS	98
4	LINNETT	79
5	JAYDEN	45
6	SAM	63
7	ZOOEY	82
8	ROBB	97
9	JON	38
10	SANSA	54
11	ARYA	78

SELECT & WHERE:-

1. Retrieve all data from the table ‘Science_Class’



```
SELECT * FROM Science_Class;
```

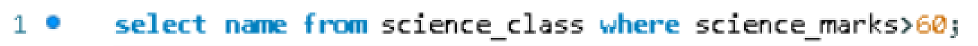


ENROLLMENT_NO	NAME	SCIENCE_MARKS
1	POPEYE	33
2	OLIVE	54
3	BRUTUS	98
4	LINNETT	79
5	JAYDEN	45
6	SAM	63
7	ZOOEY	82
8	ROBB	97
9	JON	38
10	SANSA	54
11	ARYA	78

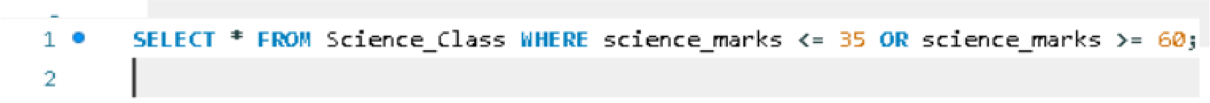
Assignment:-03

Introduction to sql and sql commands

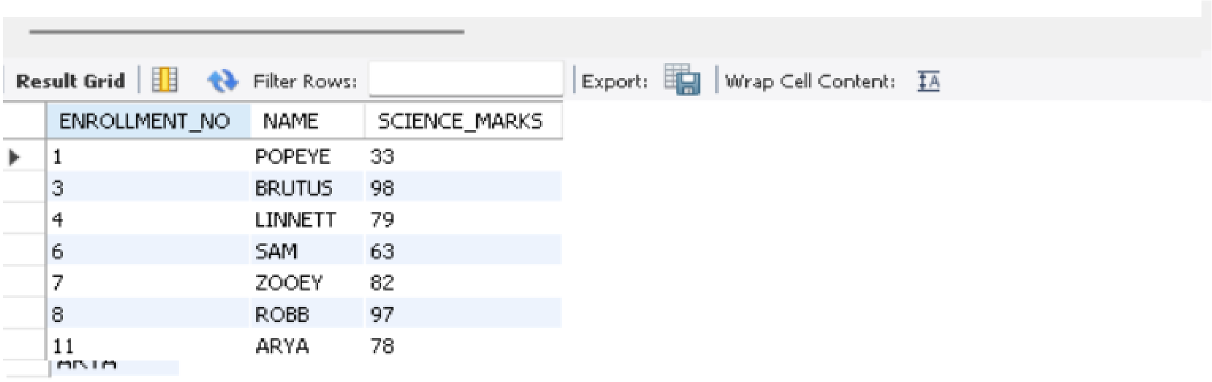
2. Retrieve the name of students who have scored more than 60 marks



```sql
select name from science_class where science_marks>60;
```

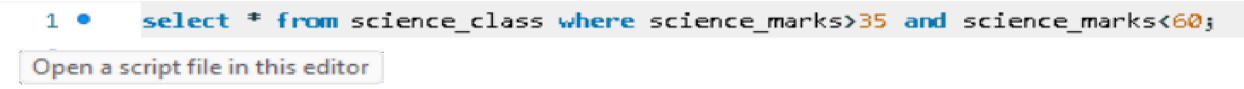


```sql
SELECT * FROM Science_Class WHERE science_marks <= 35 OR science_marks >= 60;
```

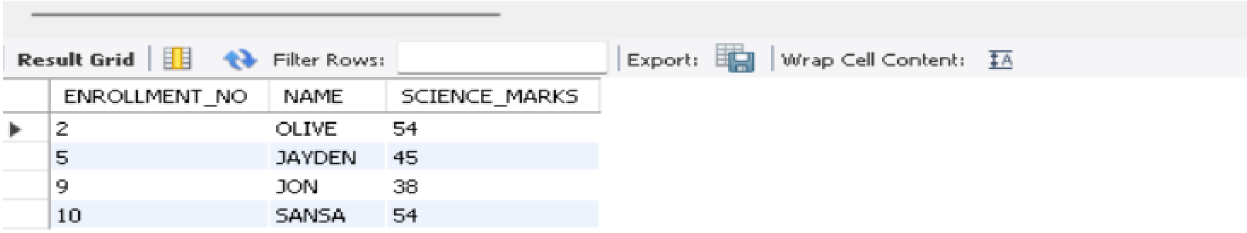


ENROLLMENT_NO	NAME	SCIENCE_MARKS
1	POPEYE	33
3	BRUTUS	98
4	LINNETT	79
6	SAM	63
7	ZOOEY	82
8	ROBB	97
11	ARYA	78

3. Retrieve all data of students who have scored more than 35 but less than 60 marks



```sql
select * from science_class where science_marks>35 and science_marks<60;
```



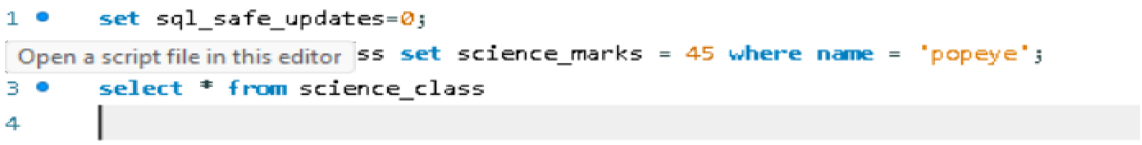
ENROLLMENT_NO	NAME	SCIENCE_MARKS
2	OLIVE	54
5	JAYDEN	45
9	JON	38
10	SANSA	54

4. Retrieve all other students i.e. who have scored less than or equal to 35 or more than or equal to 60.

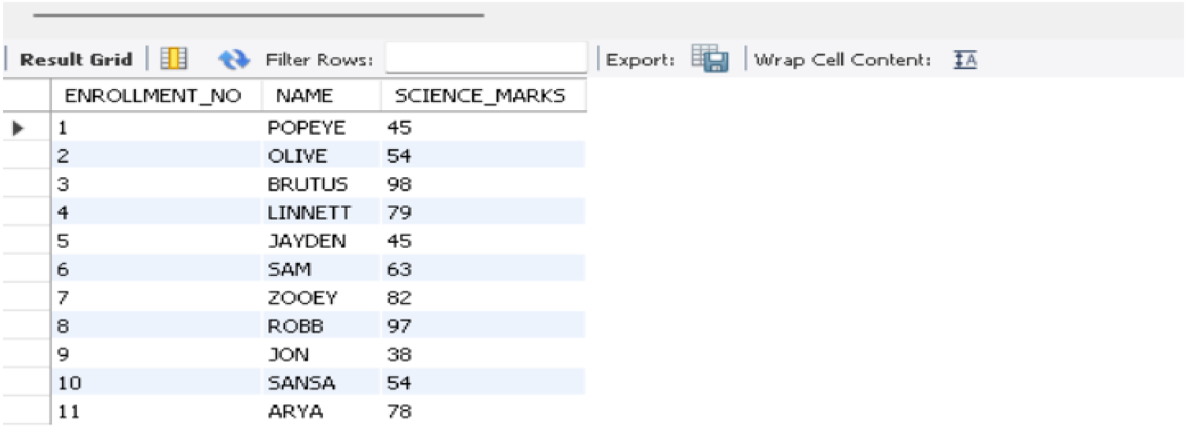
Assignment:-03
Introduction to sql and sql commands

UPDATING TABLES:-

1. Update the marks of popeye to 45



```
set sql_safe_updates=0;
ss set science_marks = 45 where name = 'popeye';
select * from science_class
```

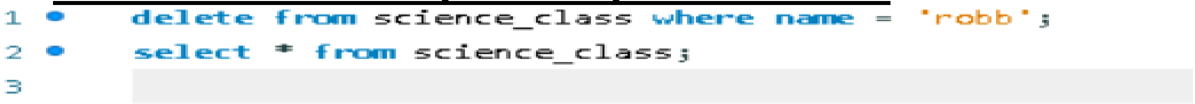


ENROLLMENT_NO	NAME	SCIENCE_MARKS
1	POPEYE	45
2	OLIVE	54
3	BRUTUS	98
4	LINNETT	79
5	JAYDEN	45
6	SAM	63
7	ZOOEY	82
8	ROBB	97
9	JON	38
10	SANSA	54
11	ARYA	78

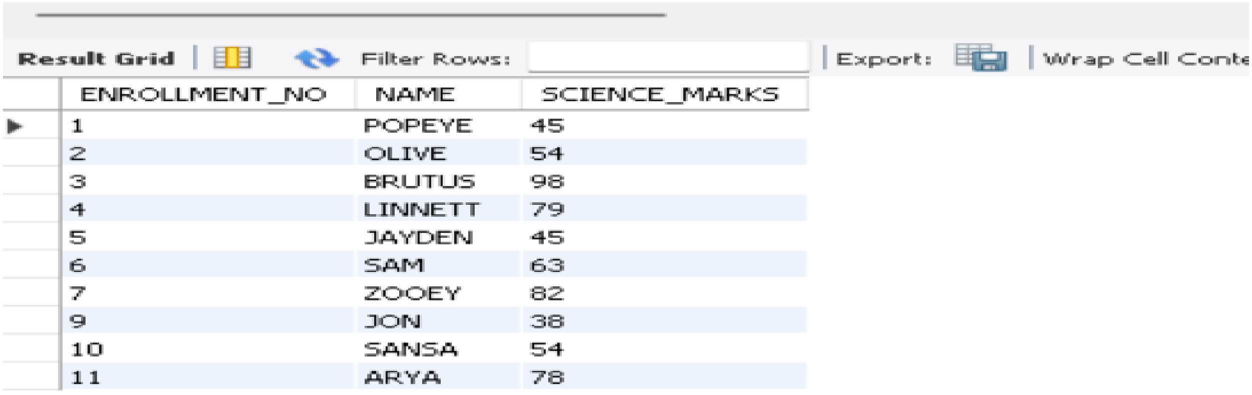
2. Delete the row containing details of the student named ‘robb’

Assignment:-03

Introduction to sql and sql commands

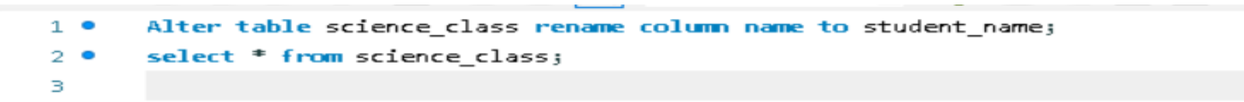


```
delete from science_class where name = 'robb';
select * from science_class;
```

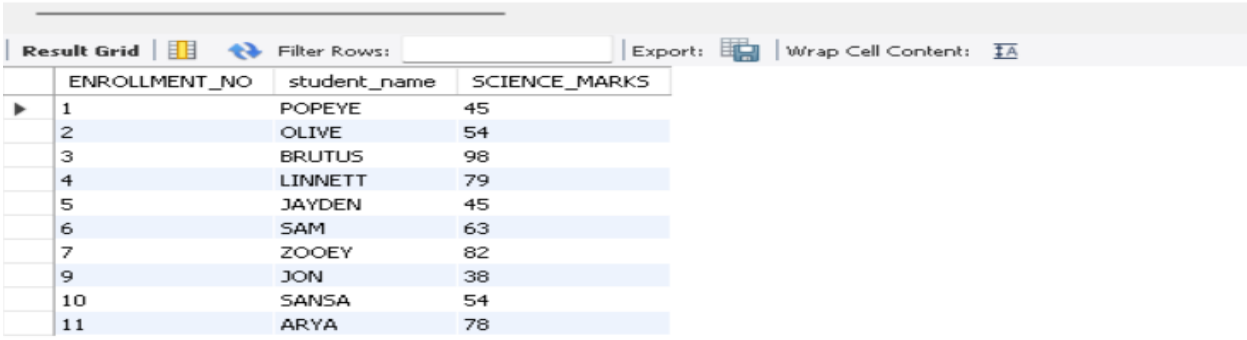


ENROLLMENT_NO	NAME	SCIENCE_MARKS
1	POPEYE	45
2	OLIVE	54
3	BRUTUS	98
4	LINNETT	79
5	JAYDEN	45
6	SAM	63
7	ZOOEY	82
9	JON	38
10	SANSA	54
11	ARYA	78

3. Rename column ‘name’ to ‘student_name’



```
Alter table science_class rename column name to student_name;
select * from science_class;
```



ENROLLMENT_NO	student_name	SCIENCE_MARKS
1	POPEYE	45
2	OLIVE	54
3	BRUTUS	98
4	LINNETT	79
5	JAYDEN	45
6	SAM	63
7	ZOOEY	82
9	JON	38
10	SANSA	54
11	ARYA	78