MANAGING TABLE

- **Create Database:**

  CREATE DATABASE databasename;

- **Create TABLE:**

```
CREATE TABLE Bugresistance (
    StudentID int,
    StudentName varchar(255),
    City varchar(255),
    PostalCode varchar(255),
    INCOME varchar(255)
    );
```

- **INSERT VALUE in Table:**

  INSERT INTO Bugresistance (StudentID, StudentName, City, PostalCode, INCOME)
  VALUES (101, 'Masud', 'Noakhali', '4000', '10000');

- **DROP TABLE:**

  DROP TABLE table_name;

- **ALTER TABLE:**

  ALTER TABLE Bugresistance ADD CGPA varchar(255);

- **NOT NULL:**

  ALTER TABLE Bugresistance ADD AGE int NOT NULL;

- **UNIQUE:**

  ALTER TABLE Bugresistance ADD NationalID int UNIQUE;

- **TRUNCATE:**

  TRUNCATE Bugresistance;

QUERYING DATA FROM SINGLE TABLE

SELECT statement is used to select data from a database

SELECT * FROM Bugresistance;

SELECT DISTINCT statement is used to return only distinct (different) values

SELECT DISTINCT City FROM Bugresistance;

SELECT COUNT(DISTINCT City) FROM Bugresistance;

WHERE clause is used to filter records

SELECT * FROM Bugresistance WHERE City='Noakhali';

SELECT * FROM Bugresistance WHERE StudentID=101;

WHERE clause can be combined with AND, OR, and NOT operators

SELECT * FROM Bugresistance WHERE City='Chittagong' AND AGE=25;

SELECT * FROM Bugresistance WHERE City='Barisal' OR AGE=30;

SELECT * FROM Bugresistance WHERE NOT City='Comilla';

SELECT * FROM Bugresistance WHERE City='Chittagong' AND (CGPA>=3.00 OR AGE=25);

ORDER BY keyword is used to sort the result-set in ascending or descending order

SELECT * FROM Bugresistance ORDER BY City;

UPDATE statement is used to modify the existing records in a table

UPDATE Bugresistance SET StudentName = 'Mohammad Ali', City= 'Hatiya' WHERE CustomerID = 104;

DELETE statement is used to delete existing records in a table

DELETE FROM Bugresistance WHERE StudentName='Jamal';

BETWEEN operator selects values within a given range

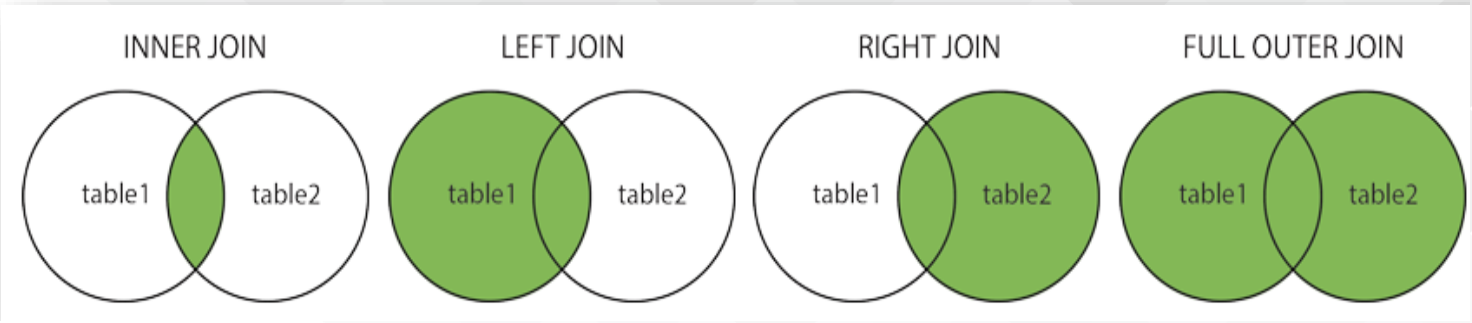
SELECT * FROM Bugresistance WHERE Income BETWEEN 10000 AND 200000;

MIN() and MAX() function returns the smallest & largest value

SELECT MIN(INCOME) FROM bugresistance;

SELECT MAX(INCOME) FROM bugresistance;

QUERYING DATA FROM MULTIPLE TABLE



INNRER JOIN

```
SELECT Bugresistance.StudentID, University.StudentName, Bugresistance.City
FROM Bugresistance
INNER JOIN University ON Bugresistance.StudentName=University.StudentName;
```

LEFT JOIN

```
SELECT Bugresistance.StudentID, University.StudentName, Bugresistance.City
FROM Bugresistance
LEFT JOIN University ON Bugresistance.StudentName=University.StudentName;
```

RIGHT JOIN

```
SELECT Bugresistance.StudentID, University.StudentName, Bugresistance.City
FROM Bugresistance
RIGHT JOIN University ON Bugresistance.StudentName=University.StudentName;
```

FULL OUTER JOIN

```
SELECT Bugresistance.StudentID, University.StudentName, Bugresistance.City
FROM Bugresistance
FULL OUTER JOIN University ON Bugresistance.StudentName=University.StudentName;
```