SQL Assignment-Day 2

Name : Vikas Reddy Gorantla

Date : 5-11-24

USE School;

-- Create the Student table

CREATE TABLE Student (

Id INT IDENTITY PRIMARY KEY,

Name VARCHAR(65) NOT NULL,

Gender CHAR(1) NULL,

Age INT NULL,

Marks INT

);

A screenshot of a computer

Description automatically generated

-- Storing data in table

INSERT INTO Student (Name, Gender, Age, Marks) VALUES ('Alice', 'F', 20, 85);

INSERT INTO Student (Name, Gender, Age, Marks) VALUES ('Bob', 'M', 22, 90);

INSERT INTO Student (Name, Gender, Age, Marks) VALUES ('Charlie', 'M', 23, 70);

INSERT INTO Student (Name, Gender, Age, Marks) VALUES ('Diana', 'F', 21, 95);

INSERT INTO Student (Name, Gender, Age, Marks) VALUES ('Eve', 'F', 22, 88);

A screenshot of a computer

Description automatically generated

-- Update data in the table

UPDATE Student

SET Marks = 92

WHERE Name = 'Bob';

A screenshot of a computer

Description automatically generated

-- Delete data in table

DELETE FROM Student

WHERE Name = 'Charlie';

A screenshot of a computer

Description automatically generated

-- Retrieve specific columns

SELECT Name, Marks

FROM Student;

A screenshot of a computer

Description automatically generated

-- Filtering Data with IN, DISTINCT, AND, OR, BETWEEN, LIKE, Column & Table Aliases

--IN Clause

SELECT \* FROM Student

WHERE Marks IN (85, 90);

A screen shot of a computer

Description automatically generated

--DISTINCT

SELECT DISTINCT Age

FROM Student;

A screenshot of a computer

Description automatically generated

--AND,OR

SELECT \* FROM Student

WHERE (Gender = 'F' AND Age > 20) OR (Gender = 'M' AND Marks > 85);

A screenshot of a computer

Description automatically generated

--BETWEEN

SELECT \* FROM Student

WHERE Marks BETWEEN 80 AND 95;

A screenshot of a computer

Description automatically generated

--LIKE

SELECT \* FROM Student

WHERE Name LIKE 'A%';

A screenshot of a computer

Description automatically generated

-- Use aliases for columns and tables

SELECT S.Name AS StudentName, S.Marks AS Score

FROM Student AS S;

A screenshot of a computer

Description automatically generated

-- Summarizing and Grouping Data

SELECT COUNT(\*) AS TotalStudents, AVG(Marks) AS AverageMarks,

MIN(Marks) AS MinMarks, MAX(Marks) AS MaxMarks

FROM Student;

SELECT Gender, AVG(Marks) AS AverageMarks FROM Student GROUP BY Gender;

A screenshot of a computer program

Description automatically generated

-- Filtering Data

SELECT \* FROM Student WHERE Marks > 85 AND Age < 25;

A screenshot of a computer

Description automatically generated

-- Total Aggregations

SELECT COUNT(\*) AS TotalStudents, AVG(Marks) AS AverageMarks FROM Student;

A close up of a screen

Description automatically generated

-- Group By Aggregations

SELECT Gender, COUNT(\*) AS NumberOfStudents, AVG(Marks) AS AverageMarks

FROM Student GROUP BY Gender;

A screenshot of a computer

Description automatically generated

-- Order of Execution

SELECT Gender, AVG(Marks) AS AverageMarks FROM Student

WHERE Age >= 20 GROUP BY Gender HAVING AVG(Marks) > 70 ORDER BY AverageMarks DESC;

A screen shot of a computer

Description automatically generated

-- Group & Filter Rules

SELECT Gender, AVG(Marks) AS AverageMarks FROM Student GROUP BY Gender;

A close-up of a website

Description automatically generated

-- Filter with Group By and Having

SELECT Gender, AVG(Marks) AS AverageMarks FROM Student GROUP BY Gender HAVING AVG(Marks) > 80;

A close-up of a computer screen

Description automatically generated