**Single-Row Subqueries:**

1. SELECT TrackId, name,

(SELECT max(Milliseconds) FROM tracks)

FROM tracks

1. SELECT TrackId, name,

(SELECT min(Milliseconds) FROM tracks)

FROM tracks

1. SELECT TrackId, name, Bytes,

(SELECT avg(Bytes) FROM tracks)

FROM tracks

WHERE Bytes > (SELECT avg(Bytes) from tracks)

ORDER BY Bytes DESC

**Multiple-Row Subqueries:**

1. SELECT CustomerId, FirstName, LastName

FROM customers

WHERE SupportRepId

IN

(SELECT EmployeeId

FROM employees

WHERE EmployeeId

IN (3,4))

1. SELECT c.CustomerId, c.FirstName, c.LastName

FROM customers c

INNER JOIN employees e

ON c.SupportRepId = e.EmployeeId

WHERE EmployeeId

IN (3,4)

**DDL (CREATE, ALTER, DELETE) and DML (SELECT, INSERT, UPDATE, DELETE) Statements**

1. CREATE TABLE courses(

CourseId INT PRIMARY KEY,

CourseName TEXT NOT NULL,

CoursePrice REAL,

EmployeeId INT,

FOREIGN KEY (EmployeeId)

REFERENCES employees(EmployeeId)

);

1. INSERT INTO courses (CourseId, CourseName, CoursePrice, EmployeeId)

VALUES (1, 'AWS', '120', 2),

(2, 'Data Science', '150', 3),

(3, 'SQL', '50', 4),

(4, 'HTML', '100', 2),

(5, 'CSS', '100', 1),

(6, 'JAVA', '150', 4),

(7, 'JavaScript', '120', 8),

(8, 'AGILE', '50', 6),

(9, 'GIT', '50', 5),

(10, 'GitHub', '50', 7);

1. DELETE FROM courses

WHERE CourseId = 10;

1. ALTER TABLE courses

ADD COLUMN StartDate;

1. ALTER TABLE courses

DROP COLUMN CoursePrice;

1. DROP TABLE courses;