Lab 10:

Create table Sales and Products using key constraints (primary key and foreign key), insert suitable data and perform DML operations (SELECT CLAUSE, WHERE CLAUSE, AGGREGATE Functions).

SQL Query:

* CREATE DATABASE db10;

USE db10;

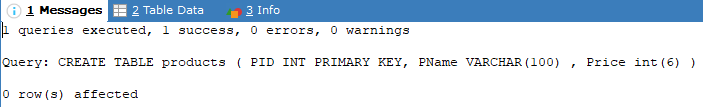
* CREATE TABLE products (

PID INT PRIMARY KEY,

PName VARCHAR(100) ,

Price INT(6)

);



* CREATE TABLE Sales (

SID INT PRIMARY KEY,

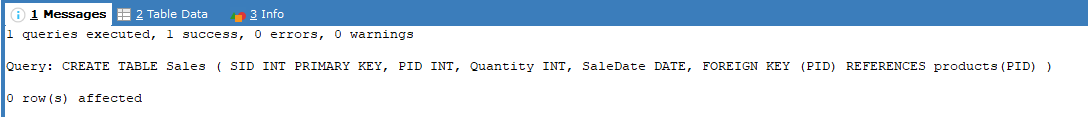
PID INT,

Quantity INT,

SaleDate DATE,

FOREIGN KEY (PID) REFERENCES products(PID)

);



* INSERT INTO products (PID, PName, Price) VALUES

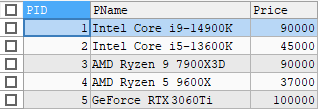
(1, 'Intel Core i9-14900K', 90000),

(2, 'Intel Core i5-13600K', 45000),

(3, 'AMD Ryzen 9 7900X3D ', 90000),

(4, 'AMD Ryzen 5 9600X', 37000),

(5, 'GeForce RTX 3060Ti', 100000);



* INSERT INTO sales (SID, PID, Quantity, SaleDate) VALUES

(101, 1, 5, '2025-08-13'),

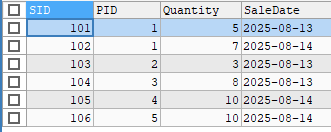
(102, 1, 7, '2025-08-14'),

(103, 2, 3, '2025-08-13'),

(104, 3, 8, '2025-08-13'),

(105, 4, 10, '2025-08-14'),

(106, 5, 10, '2025-08-14');



* SELECT

S.SID,

P.PName,

S.Quantity,

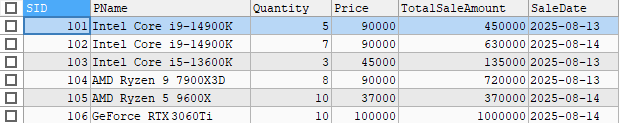
P.Price,

(S.Quantity \* P.Price) AS TotalSaleAmount,

S.SaleDate

FROM sales S

JOIN products P ON S.PID = P.PID;



* SELECT

S.SID,

P.PName,

S.Quantity,

S.SaleDate

FROM Sales S

JOIN products P ON S.PID = P.PID

WHERE P.PName = 'Intel Core i5-13600K';



* SELECT

SUM(S.Quantity \* P.Price) AS TotalRevenue

FROM sales S

JOIN products P ON S.PID = P.PID;



* SELECT

AVG(Quantity) AS AvgQuantityPerSale

FROM Sales;



* SELECT

P.PName,

COUNT(S.SaleID) AS TotalSales

FROM sales S

JOIN products P ON S.PID = P.PID

GROUP BY P.PName;

