Assignment 3:

-- create a table

CREATE TABLE EmployeeDetails (

EmpId INTEGER PRIMARY KEY,

Fullname TEXT NOT NULL,

ManagerId INTEGER,

DateOfJoining date,

City TEXT

);

CREATE TABLE EmployeeSalary (

EmpId INTEGER PRIMARY KEY,

Project TEXT NOT NULL,

Salary INTEGER,

Variable INTEGER

);

-- insert values in EmplyeeDetails

INSERT INTO EmployeeDetails VALUES (121, 'John Snow', 321, '01/31/2014','Toronto');

INSERT INTO EmployeeDetails VALUES (321, 'Walter White', 986, '01/30/2015','Toronto');

INSERT INTO EmployeeDetails VALUES (421, 'Kuldeep Rana', 876, '27/11/2016','Toronto');

-- insert values in EmplyeeSalary

INSERT INTO EmployeeSalary VALUES (121, 'P1', 8000,500);

INSERT INTO EmployeeSalary VALUES (321, 'P2', 10000,1000);

INSERT INTO EmployeeSalary VALUES (421, 'P1', 12000,0);

-- fetch some values

SELECT \* FROM EmployeeDetails;

-- -- fetch some values

SELECT \* FROM EmployeeSalary;

-- A. Write an SQL query to display the total salary of each employee adding the Salary with Variable

-- value

SELECT EmployeeDetails.\*, (EmployeeSalary.Salary + EmployeeSalary.Variable) AS TotalSalary

FROM EmployeeDetails

LEFT JOIN EmployeeSalary ON EmployeeDetails.EmpId = EmployeeSalary.EmpId;

-- B. Write an SQL query to fetch employee names having a salary greater than or equal to 5000 and

-- less than or equal to 10000.

SELECT EmployeeDetails.Fullname

FROM EmployeeDetails

JOIN EmployeeSalary ON EmployeeDetails.EmpId = EmployeeSalary.EmpId

WHERE EmployeeSalary.Salary BETWEEN 5000 AND 10000;

-- C. Write a query to fetch employee names and salary records. Display the employee details even if

-- the salary record is not present for the employee.

SELECT EmployeeDetails.Fullname, EmployeeSalary.Salary

FROM EmployeeDetails

LEFT JOIN EmployeeSalary ON EmployeeDetails.EmpId = EmployeeSalary.EmpId;