ADBMS

Lab sheet 1

**Part 1**

1. Create a Database in SQL Server to save details about office activities.

**DB: OfficeDB**

1. Use OfficeDB and Create following table structures in SQL Server by adding suitable data types, field sizes and Constraints.

(Identify the relevant constraints and give constraint names. Ex: primary key, foreign key etc.)

1. Input data to the tables as below.

**Employee**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Fname | Lname | Emp\_id | DOB | Salary | Gender | address | Dep\_no | Designation |
| Maria | Smith | 10011 | 1972-02-03 | 20000 | F | 231, Los Angeles, California | 1 | Manager |
| Soobin | Choi | 10012 | 1963-05-09 | 50000 | M | 112, San Diego, California | 2 | Executive |
| kai | James | 10024 | 1995-01-18 | 30000 | M | 143, Oakland, California | 2 | Manager |
| John | Wang | 10013 | 1991-06-21 | 10000 | M | 085, castle, Texas | 3 | Cashier |
| Lisa | Ash | 10032 | 1991-08-09 | 10500 | F | 022, Houston, Texas | 3 | Cashier |
| Jennie | Kim | 10045 | 1980-11-13 | 40000 | F | 311, Texas city, Texas | 3 | Manager |

**Department**

|  |  |  |  |
| --- | --- | --- | --- |
| Dep\_name | Dnumber | Manager\_id | Manager\_start\_date |
| HQ | 1 | 10011 | 2020-01-14 |
| Admin | 2 | 10024 | 2020-03-15 |
| Production | 3 | 10045 | 2020-03-24 |

**Dept\_location**

|  |  |
| --- | --- |
| Dnumber | Dlocation |
| 1 | Texas |
| 2 | LA |
| 3 | Oakland |
| 3 | Texas |

**Project**

|  |  |  |  |
| --- | --- | --- | --- |
| Pr\_name | Pr\_no | Pr\_location | Dep\_no |
| Pro\_A | 1 | Beverly Hills | 1 |
| Pro\_B | 2 | Houston | 1 |
| Pro\_C | 3 | Los Angeles | 1 |
| Pro\_D | 15 | San Francisco | 2 |
| Pro\_E | 10 | Los Angeles | 3 |
| Pro\_F | 9 | Houston | 3 |

**Works\_on**

|  |  |  |
| --- | --- | --- |
| Empid | Pnumber | hours |
| 10011 | 1 | 32.5 |
| 10010 | 3 | 40.0 |
| 10032 | 2 | 10.0 |
| 10045 | 10 | 10.0 |
| 10099 | 10 | NULL |
| 10011 | 15 | 8.5 |

**Dependent**

|  |  |  |  |
| --- | --- | --- | --- |
| Emp\_id | Dependent\_name | DOB | Relationship |
| 10012 | Rose | 1997-08-08 | Daughter |
| 10012 | Billie | 1995-07-02 | Son |
| 10013 | Selena | 1990-03-04 | Spouse |
| 10011 | Jimin | 1991-11-14 | Son |

**Supervisor**

|  |  |  |
| --- | --- | --- |
| SupervisorID | Supervisorname | Employeeid |
| 7 | Jeremy | 10013 |
| 8 | Anne | 10032 |

**Part 2**

**Using Sub queries find the outputs for the followings,**

1. Display employee ID, first name and last name of all the employees who work in the same department where employee who has the ID “10045” is working.
2. Display Emp id, first name and department no of all the employees who work in departments located at ‘’Texas”.
3. Retrieve the names of the employees who do not have Dependents.
4. The departments that has two employees or more than two employees, display the department number and number of the employees in that department who has a salary more than 10,500.
5. Get the names of the employees who has more than one dependent.

**Part 3**

**Using Suitable Joins find the outputs for the followings,**

1. Retrieve the number of Employees in the ‘’Production’’ department.
2. Display the employee name, employee ID, dependent name and relationship of the dependent with the employee, in one result set.
3. Combine and display the details from the tables Employee and dependent, but there should not be any duplicate rows.
4. Display the Department name, Project location, Project name and Employee ID of the Employees who works on projects all as one result set and sort the table according to the Descending order of the Project location.
5. Display all the details of the Employee table and all the details of the Works\_on table together in one result set.
6. Display Employee name and the Designation of the employees who has a supervisor along with the supervisor’s Details. (Every employee may not have a supervisor).