|  |
| --- |
| **www.enosislearning.com** |
| 4.0 SQL ASSIGNMENTS |
| This assignments is related to Foreign Key & Relational Database Management system of SQL SERVER. |

|  |
| --- |
|  |

**CREATE A TABLE FOR COMPANY**

|  |  |  |
| --- | --- | --- |
| COLUMN\_NAME | DATATYPE | CONSTRAINT |
| COMPANYID | INT | PRIMARY KEY, IDENTITY(1,1) |
| COMPANYNAME | VARCHAR(100) |  |
| WEBSITE | VARCHAR(100) |  |
| EMAIL | VARCHAR(100) |  |

**ADD TWO COMPANY NAME TO THE TABLE using INSERT STATEMENTS**

**CREATE A TABLE FOR DEPT**

|  |  |  |
| --- | --- | --- |
| COLUMN\_NAME | DATATYPE | CONSTRAINT |
| DEPTID | INT | PRIMARY KEY, IDENTITY(1,1) |
| DEPTNAME | VARCHAR(100) |  |
| DEPTLOCATION | VARCHAR(100) |  |
|  |  |  |

**ADD FIVE DEPT NAME TO THE TABLE**

**CREATE A TABLE FOR DESG**

|  |  |  |
| --- | --- | --- |
| COLUMN\_NAME | DATATYPE | CONSTRAINT |
| DESGID | INT | PRIMARY KEY |
| DESGNAME | VARCHAR(100) |  |
| CREATEDDATE | DATETIME | DEFAULT - GETDATE() |
|  |  |  |

**ADD FIVE DESIGNATION NAMES TO THE TABLE**

**CREATE A TABLE FOR EMPLOYEE**

|  |  |  |
| --- | --- | --- |
| COLUMN\_NAME | DATATYPE | CONSTRAINT |
| EMPID | INT | PRIMARY KEY |
| EMPNAME | VARCHAR(100) | UNIQUE |
| BASICSALARY | FLOAT |  |
| INCENTIVE | FLOAT |  |
| BONUS | FLOAT |  |
| DOB | DATETIME |  |
| DOJ | DATETIME | DEFAULT GETDATE() |
| GENDER | VARCHAR(100) |  |
| EMAIL | VARCHAR(100) | UNIQUE |
| MOBILE | VARCHAR(10) |  |
| CREATEDDATE | DATETIME | DEFAULT GETDATE() |
| COMPANYID | INT | F.K |
| DEPTID | INT | F.K |
| DESGID | INT | F.K |

**ADD 10 EMPLOYEES TO THE TABLE**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **WRITE A QUERY TO DISPLAY THE TOTALSALARY FOR EACH DEPTNAME.**   |  |  | | --- | --- | | DEPTNAME | TOTALSALARY | |  |  |   **WRITE A QUERY TO DISPLAY THE NOOFEMPLOYEES FOR EACH COMPANYNAME.'**   |  |  | | --- | --- | | COMPANYNAME | NOOFEMPLOYEES | |  |  |   **WRITE A QUERY TO DISPLAY THE DESGNAME,NOOFEMPS,TOTALSALARY**   |  |  |  | | --- | --- | --- | | DESGNAME | NOOFEMPS | TOTALSALARY | |  |  |  | |

**CREATE A TABLE COUNTRY AND ADD THE VALUES AS SHOWN BELOW**

|  |  |  |
| --- | --- | --- |
| COUNTRYID(P.K) | COUNTRYNAME(UNIQUE) | COUNTRYCAPITAL |
| 1 | INDIA | DELHI |
| 2 | CHINA | BEIJING |
| 3 | U.S.A | Washington |

**CREATE A TABLE STATE AND ADD THE VALUES AS SHOWN BELOW**

|  |  |  |  |
| --- | --- | --- | --- |
| STATEID(P.K) | STATENAME | STATECAPITAL | COUNTRYID |
| 1 | MH | MUMBAI | 1 |
| 2 | MP | BHOPAL | 1 |
| 3 | UP | LUCKNOW | 1 |
| 4 | KARNATAKA | BANGALORE | 1 |
| 5 | Alaska | Juneau | 3 |
| 6 | New York | Albany | 3 |
| 7 | Berlin | Berlin | NULL |

**CREATE A TABLE CITY AND ADD THE VALUES AS SHOWN BELOW**

|  |  |  |
| --- | --- | --- |
| CITYID(P.K) | CITYNAME | STATEID |
| 1 | NASHIK | 1 |
| 2 | PUNE | 1 |
| 3 | NAGPUR | 1 |

WRITE THE QUERIES USING THE COUNTRY, STATES AND CITY TABLE

|  |
| --- |
| 1. WRITE A QUERY TO DO An INNER JOIN BETWEEN COUNTRY AND STATE TABLE. 2. WRITE A QUERY TO DO An LEFT OUTER JOIN BETWEEN COUNTRY AND STATE TABLE. 3. WRITE A QUERY TO DO An RIGHT OUTER JOIN BETWEEN COUNTRY AND STATE TABLE. 4. WRITE A QUERY TO DO An FULL OUTER OUTER JOIN BETWEEN COUNTRY AND STATE TABLE. 5. WRITE A QUERY TO DO A CROSS JOIN BETWEEN COUNTRY AND STATE TABLE 6. WRITE A QUERY TO FIND THE NAME OF COUNTRIES WHICH ARE NOT PRESENT IN THE INNER JOIN. (HINT : USE LEFT JOIN EXCEPT INNER JOIN) |

**ADD A COLUMN TO EMPLOYEE TABLE CITYID WITH A FOREIGN KEY TO CITY TABLE**

**WRITE A QUERY TO FIND THE NUMBER OF EMPLOYEES FOR EACH CITYNAME**

|  |  |
| --- | --- |
| CITYNAME | NOOFEMPS |
|  |  |

**WRITE A QUERY TO FIND THE NUMBER OF STATES IN EACH COUNTRY**

|  |  |
| --- | --- |
| COUNTRYNAME | NOOFSTATES |
|  |  |
|  |  |

**WRITE A QUERY TO FIND HIGHEST SALARY HOLDER IN EMPLOYEE TABLE**

**WRITE A QUERY TO FIND 3RD HIGHEST SALARY HOLDER IN EMPLOYEE TABLE**

**WRITE A QUERY TO FIND FIFTH HIGHEST SALARY HOLDER IN EMPLOYEE TABLE**

SELF JOIN ASSIGNMENT

**CREATE A TABLE EMPLOYEE AND ADD THE VALUES AS SHOWN BELOW**

**IN THE EXAMPLE, THE MANAGERID WILL BE FOREIGN KEY WHICH WILL HAVE REFERENCE TO THE PRIMARY KEY EMPID (**PRIMARY KEY)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| EMPID (PRIMARY KEY) | EMPNAME | SALARY | TAX | MANAGERID (FOREIGN KEY) |
| 1 | SAMIR | 98000 | 1000 | NULL |
| 2 | YOGESH | 88000 | 2000 | 1 |
| 3 | NEERAJ | 58000 | 2000 | 1 |
| 4 | KAMLESH | 48000 | 1000 | 2 |

WRITE A QUERY TO FIND THE MANAGERNAME OF THE EMPLOYEES.

OUTPUT WILL BE

|  |  |  |  |
| --- | --- | --- | --- |
| EMPNAME | EMPSALARY | MANAGERNAME | MANAGER\_SALARY |
|  |  |  |  |
|  |  |  |  |