**Class Test 01**

1. Display the job and the hire date of the most senior employee of each department and order the result by descending order.

**Answer: SELECT JOB, HIREDATE**

**FROM EMP**

**WHERE (DEPTNO, HIREDATE) IN (**

**SELECT DEPTNO, MAX(HIREDATE)**

**FROM EMP**

**GROUP BY DEPTNO**

**)**

**ORDER BY HIREDATE DESC;**

1. Display today’s day and time. The format must return a result like this: Sunday 3:30:45.Label the column as Day and Time.

**Answer: SELECT TO\_CHAR(SYSDATE, 'Day HH:MI:SS') AS "Day and Time"**

**FROM DUAL;**

1. Display the name, salary and dream salary. If department is 10 dream salary will be actual salary+1000, if department is 20 dream salary will be actual salary+2000. For all other departments the dream salary will be actual salary+3000. Sort by ascending order of dream salary.

**Answer: SELECT ENAME, SAL,**

**CASE DEPTNO**

**WHEN 10 THEN SAL + 1000**

**WHEN 20 THEN SAL + 2000**

**ELSE SAL + 3000**

**END AS "Dream Salary"**

**FROM EMP**

**ORDER BY "Dream Salary" ASC;**

1. Find out if the year 2016 is a leap year or not.

**Answer: SELECT**

**CASE**

**WHEN MOD(2016, 4) = 0 AND (MOD(2016, 100) != 0 OR MOD(2016, 400) = 0)**

**THEN 'Leap Year'**

**ELSE 'Not a Leap Year'**

**END AS Status FROM dual;**

1. Display the name and job of the employee where the employee is a manager or clerk

or analyst. You must use IN operator.

**Answer: SELECT ENAME, JOB**

**FROM EMP**

**WHERE JOB IN ('MANAGER', 'CLERK', 'ANALYST');**

1. Display only the names of employees who have at least one M.

**Answer: SELECT ENAME**

**FROM EMP**

**WHERE ENAME LIKE '%M%';**

1. Display the name of all employees who have two *L’*s or two T’s in their name. Make sure there is no other letters in between the said letters.

**Answer: SELECT ENAME**

**FROM EMP**

**WHERE ENAME LIKE '%LL%' OR ENAME LIKE '%TT%';**

1. Trim the string 00abc000 in such a way that the output is 00abc.

**Answer: SELECT CONCAT('00', TRIM('0' FROM '00abc000')) AS trimmed\_string**

**FROM DUAL;**

|  |  |
| --- | --- |
| **t\_id** | Number(10) |
| **t\_name** | Varchar2(20) |
| **phone** | Number(10) |
| **salary** | Number(7) |
| **email** | Varchar2(30) |

1. Create a table named **Teacher** from following structure using SQL command.

**Answer: CREATE TABLE Teacher (**

**t\_id NUMBER(10),**

**t\_name VARCHAR2(20),**

**phone NUMBER(10),**

**salary NUMBER(7),**

**email VARCHAR2(30)**

**);**

**Subquery**

1. Display the employee names who joined after KING.

**Answer: SELECT ENAME**

**FROM EMP**

**WHERE HIREDATE > (**

**SELECT HIREDATE**

**FROM EMP**

**WHERE ENAME = 'KING'**

**);**

1. Display the employee names that earn a salary that is higher than the salary of all CLERKS.

**Answer: SELECT ENAME**

**FROM EMP**

**WHERE SAL > (**

**SELECT MAX(SAL)**

**FROM EMP**

**WHERE JOB = 'CLERK'**

**);**

1. Display the employee names that earn a salary that is higher than the salary of any CLERKS.

**Answer: SELECT ENAME**

**FROM EMP**

**WHERE SAL > ANY (**

**SELECT SAL**

**FROM EMP**

**WHERE JOB = 'CLERK'**

**);**

1. Display the employee names who get the department wise highest amount of salary.

**Answer: SELECT ENAME**

**FROM EMP**

**WHERE (DEPTNO, SAL) IN (**

**SELECT DEPTNO, MAX(SAL)**

**FROM EMP**

**GROUP BY DEPTNO**

**);**

1. Display the employee names who earn more than employee SMITH.

**Answer: SELECT ENAME**

**FROM EMP**

**WHERE SAL > (**

**SELECT SAL**

**FROM EMP**

**WHERE ENAME = 'SMITH'**

**);**

**Joining**

1. Display the name of all the employees who work in DALLAS.

**Answer: SELECT E.ENAME**

**FROM EMP E**

**JOIN DEPT D ON E.DEPTNO = D.DEPTNO**

**WHERE D.LOC = 'DALLAS';**

1. Display the name of all the employees who joined before their manager.

**Answer: SELECT E.ENAME**

**FROM EMP E**

**JOIN EMP M ON E.MGR = M.EMPNO**

**WHERE E.HIREDATE < M.HIREDATE;**

1. Display the name of all the employees and their respective manager. If an employee does not have a manger display ‘NO MANAGER’.

**Answer: SELECT E.ENAME, NVL(M.ENAME, 'NO MANAGER') AS MANAGER**

**FROM EMP E**

**LEFT JOIN EMP M ON E.MGR = M.EMPNO;**

1. Display the salary grades of each employee.

**Answer:** **SELECT E.ENAME, S.GRADE**

**FROM EMP E**

**JOIN SALGRADE S ON E.SAL BETWEEN S.LOSAL AND S.HISAL;**

1. Write a query to display the name, department number, and department name for  
   all employees.

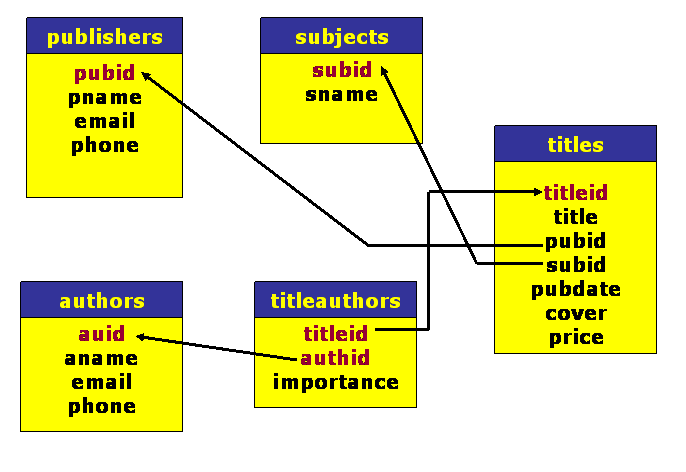
**Answer:** **SELECT E.ENAME, E.DEPTNO, D.DNAME**

**FROM EMP E**

**JOIN DEPT D ON E.DEPTNO = D.DEPTNO;**

**Schema Diagram**

1. Create the tables according to the schema provided below. Arrows point from foreign key to corresponding primary key. Also insert 5 rows of data into each table created.



**Answer**:

**-- Create the publishers table**

**CREATE TABLE publishers (**

**pubid NUMBER PRIMARY KEY,**

**pname VARCHAR2(20),**

**email VARCHAR2(20),**

**phone NUMBER**

**);**

**-- Create the subjects table**

**CREATE TABLE subjects (**

**subid NUMBER PRIMARY KEY,**

**sname VARCHAR2(20)**

**);**

**-- Create the titles table**

**CREATE TABLE titles (**

**titleid NUMBER PRIMARY KEY,**

**title VARCHAR2(20),**

**pubid NUMBER,**

**subid NUMBER,**

**pubdate DATE,**

**cover VARCHAR2(100),**

**price NUMBER,**

**CONSTRAINT fk\_titles\_publishers FOREIGN KEY (pubid) REFERENCES publishers (pubid),**

**CONSTRAINT fk\_titles\_subjects FOREIGN KEY (subid) REFERENCES subjects (subid)**

**);**

**-- Create the authors table**

**CREATE TABLE authors (**

**authid NUMBER PRIMARY KEY,**

**aname VARCHAR2(20),**

**email VARCHAR2(20),**

**phone NUMBER**

**);**

**-- Create the titleauthors table**

**CREATE TABLE titleauthors (**

**titleid NUMBER,**

**authid NUMBER,**

**importance VARCHAR2(20),**

**CONSTRAINT fk\_titleauthors\_titles FOREIGN KEY (titleid) REFERENCES titles (titleid),**

**CONSTRAINT fk\_titleauthors\_authors FOREIGN KEY (authid) REFERENCES authors (authid)**

**);**

**INSERT ALL**

**INTO publishers (pubid, pname, email, phone) VALUES (1, 'Publisher A', 'publisherA@exale.com', '1234567890')**

**INTO publishers (pubid, pname, email, phone) VALUES (2, 'Publisher B', 'publisherB@emple.com', '9876543210')**

**INTO publishers (pubid, pname, email, phone) VALUES (3, 'Publisher C', 'publisherC@ample.com', '5555555555')**

**INTO publishers (pubid, pname, email, phone) VALUES (4, 'Publisher D', 'publisherD@ample.com', '1111111111')**

**INTO publishers (pubid, pname, email, phone) VALUES (5, 'Publisher E', 'publisherE@ample.com', '9999999999')**

**SELECT 1 FROM DUAL;**

**INSERT ALL**

**INTO subjects (subid, sname) VALUES (1, 'Subject A')**

**INTO subjects (subid, sname) VALUES (2, 'Subject B')**

**INTO subjects (subid, sname) VALUES (3, 'Subject C')**

**INTO subjects (subid, sname) VALUES (4, 'Subject D')**

**INTO subjects (subid, sname) VALUES (5, 'Subject E')**

**SELECT 1 FROM DUAL;**

**INSERT ALL**

**INTO titles (titleid, title, pubid, subid, pubdate, cover, price) VALUES (1, 'Title A', 1, 1, TO\_DATE('2023-01-01', 'YYYY-MM-DD'), 'Cover A', 10)**

**INTO titles (titleid, title, pubid, subid, pubdate, cover, price) VALUES (2, 'Title B', 2, 1, TO\_DATE('2023-02-01', 'YYYY-MM-DD'), 'Cover B', 15)**

**INTO titles (titleid, title, pubid, subid, pubdate, cover, price) VALUES (3, 'Title C', 3, 2, TO\_DATE('2023-03-01', 'YYYY-MM-DD'), 'Cover C', 20)**

**INTO titles (titleid, title, pubid, subid, pubdate, cover, price) VALUES (4, 'Title D', 2, 3, TO\_DATE('2023-04-01', 'YYYY-MM-DD'), 'Cover D', 25)**

**INTO titles (titleid, title, pubid, subid, pubdate, cover, price) VALUES (5, 'Title E', 1, 2, TO\_DATE('2023-05-01', 'YYYY-MM-DD'), 'Cover E', 30)**

**SELECT 1 FROM DUAL;**

**INSERT ALL**

**INTO authors (authid, aname, email, phone) VALUES (1, 'Author A', 'authorA@example.com', '1111111111')**

**INTO authors (authid, aname, email, phone) VALUES (2, 'Author B', 'authorB@example.com', '2222222222')**

**INTO authors (authid, aname, email, phone) VALUES (3, 'Author C', 'authorC@example.com', '3333333333')**

**INTO authors (authid, aname, email, phone) VALUES (4, 'Author D', 'authorD@example.com', '4444444444')**

**INTO authors (authid, aname, email, phone) VALUES (5, 'Author E', 'authorE@example.com', '5555555555')**

**SELECT 1 FROM DUAL;**

**INSERT ALL**

**INTO titleauthors (titleid, authid, importance) VALUES (1, 1, 'Primary')**

**INTO titleauthors (titleid, authid, importance) VALUES (1, 2, 'Secondary')**

**INTO titleauthors (titleid, authid, importance) VALUES (2, 3, 'Primary')**

**INTO titleauthors (titleid, authid, importance) VALUES (2, 4, 'Secondary')**

**INTO titleauthors (titleid, authid, importance) VALUES (3, 5, 'Primary')**

**SELECT 1 FROM DUAL;**

**\*\*After solving the above questions using Oracle 10g, write the SQLs using pen and paper. Make sure to write your Name, ID and Signature on the paper.**