|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **EMPNO**  Number(4)  Primary Key | **ENAME**  Varchar2(10) | **JOB**  Not Null | **MGR**  Varchar2(9) | **HIREDATE**  Date | **SAL**  Number(7,2) | **COMM**  Number(7,2) | **DEPTNO**  Number(2) |
| 7839 | King | President |  | 17-Nov-01 | 50000 |  | 10 |
| 7698 | Blake | Manager | 7839 | 01-May-01 | 28500 |  | 30 |
| 7782 | Clark | Manager | 7839 | 09-Jun-01 | 24500 |  | 10 |
| 7566 | Jones | Manager | 7839 | 02-Apr-01 | 29750 |  | 20 |
| 7654 | Martin | Salesman | 7698 | 28-Sep-01 | 12500 | 14000 | 30 |
| 7499 | Allen | Salesman | 7698 | 20-Feb-01 | 16000 | 3000 | 30 |
| 7844 | Turner | Salesman | 7698 | 08-Feb-01 | 15000 | 0 | 30 |
| 7900 | James | Clerk | 7698 | 03-Dec-01 | 9500 |  | 30 |
| 7521 | Ward | Salesman | 7698 | 22-Feb-01 | 12500 | 5000 | 30 |
| 7902 | Ford | Analyst | 7566 | 03-Dec-01 | 30000 |  | NULL |
| 7369 | Smith | Clerk | 7902 | 17-Dec-00 | 8000 |  | NULL |
| 7788 | Scott | Analyst | 7566 | 09-Dec-02 | 30000 |  | 20 |
| 7876 | Adams | Clerk | 7788 | 12-Jan-03 | 11000 |  | 20 |
| 7934 | Miller | Clerk | 7782 | 23-Jan-02 | 13000 |  | NULL |

**EMPLOYEE TABLE**

|  |  |  |
| --- | --- | --- |
| **GRADE**  Number(1) | **LOSAL**  Number(7,2) | **HISAL**  Number(7,2) |
| 1 | 7000 | 12000 |
| 2 | 12001 | 14000 |
| 3 | 14001 | 20000 |
| 4 | 20001 | 30000 |
| 5 | 30001 | 99999 |

|  |  |  |
| --- | --- | --- |
| **DEPTNO**  Number(2), Not Null | **DNAME**  Varchar2(15) | **LOC**  Varchar2(15) |
| 10 | Accounting | New Delhi |
| 20 | Research | Bombay |
| 30 | Sales | Chennai |
| 40 | Operation | Kolkatta |

**DEPARTMENT TABLE SALARYGRADE TABLE**

**Queries**

1. Create the above three tables with the given constraints and insert values as provided
2. WAQ to display Employee Number, Job, Sal from Employee Table
3. WAQ to display all the details of the Employees
4. WAQ to display distinct Jobs of the Employees
5. WAQ to display all Jobs of Employees using ‘ALL’ keyword
6. WAQ to display the Names, Numbers and Salaries added with their commissions from Employees Table
7. Show the structure of the three relations
8. Display the Salaries of the Employees before and after an increment of 15 % with their names
9. WAQ to display the annual salary of Employees with their names and numbers
10. WAQ to display the details of the employees and default values (In place of Null values of commission use nvl function to show 0 )
11. WAQ to display the details of Employees in the following format. “JOHN WORKS AS MANAGER AND GETS RS 10000 EVERY MONTH”
12. WAQ to display the ename joined with the Job using || operator
13. Display the names of employees having salary > 13000
14. Give a query to display the names of employees who are not Managers
15. WAQ to display the names of Employees who are managers or analysts or clerk using IN operator
16. WAQ to display the names of employees who didn’t get any commission
17. WAQ to display the names, salaries and commission of employees who don’t get commission
18. Select the names of employees whose salary is between 15000 and 20000
19. WAQ to display the names of employees with their department names
20. WAQ to display the names of employees with their salary grades
21. Display the names of employees whose name starts with ‘S’
22. Display the names of employees whose name ends with ‘D’
23. Display the names of employees whose name holds the character ‘O’ in any place
24. WAQ to display the names of employees in the order of their salary ascending and descending separately
25. Give a query to display the names of employees whose name has only four characters
26. WAQ to display the name, job, salary and hiredate of employees where salary is between 10000 and 20000 in ascending order of their hiredate
27. WAQ to display the names of employees concatenated with their job and use alias by “NameJob” and sort them using the alias name in descending order
28. Display the names, number, job, salary of employees who do not have manager
29. Write a query to perform a calculation of the expression 4 \* 5 + 6 / 2
30. Display today’s Date
31. Display the remaining days of this month
32. WAQ to display the sum, average, standard deviation, variance, no. of records, minimum, maximum of salary in employee table using alias
33. Display the minimum and maximum salary in each job
34. Display the no. of employees in each department in ascending order of the count of the employees
35. Display the no. of employees job wise within department wise
36. Show the number of managers in each department
37. Display the names of departments which consist more than 4 employees
38. WAQ to display the department number and salary difference in each department where department is not 40 and difference is > 5000 (show in ascending order of the difference)
39. Show which positions are paid a higher than average salary
40. Select second maximum salary from employee
41. Select 4the minimum salary from employee
42. Select the name of employees who are getting minimum salary in employee table
43. Select the names of employees who are getting maximum salary in each department
44. Show details of those employees that have salary equal to any employees of department
45. Show details of those employee that have salary more than or equal to average salary
46. Select the details of departments that have at least one employee

**PL/SQL Assignments**

1. Write a PL/SQL program to print the numbers below 20
2. Write a PL/SQL program to print the sum of even numbers below 20
3. Write a PL/SQL program to read two numbers and display their sum
4. Write a PL/SQL program that reads a single digit number and displays its first three multiples in the same time
5. Write a PL/SQL program to illustrate nested blocks (the outer block should calculate and display the squares of three given numbers and inner block should display the product of these given numbers)
6. Write a PL/SQL program to accept a record in to existing employee table
7. Write a PL/SQL program to access and display the record from employee table where employee number is 7839
8. Write a PL/SQL program to find the biggest among three given numbers
9. Write a PL/SQL program to reverse a given number
10. Write a PL/SQL program to display a given multiplication table

**Views**

1. Create a view to display all the details of employees
2. Create a view to display the salaries of employees before and after an increment of 15% with their names
3. Create a view to display the names of employees who are not managers
4. Create a view for displaying the names of employees who didn’t get commission
5. Create a view to display the names of employees with department names
6. Create a view to display the name, job, sal, hiredate of employees whose sal is between 10000 and 20000 in ascending order of their hiredate

**Cursors**

1. Write a program to update employee table where sal<25000 is equalized to sal = 50000 using cursors
2. Write a cursor to update the salary details as those who have a salary less than the average salary should be updated as equal to average salary
3. Write a cursor to display the names of employees
4. Write a program to get the employee details whose dept code is 10

**PL/SQL**

1. Write a PL/SQL code to increase the salary of the employee by 5% whose salary is more than 30000. Get the employee number from user
2. Write a PL/SQL code to insert a new record in the employee table after obtaining values from the user
3. Write a PL/SQL code to obtain an empno from user. If her/his salary is less than Rs. 9000 then delete that record from the table
4. Obtain details of employee drawing minimum salary. Add his/her details into a table NEWSAL (empno, ename, salary) after incrementing salary by Rs. 7000
5. Write a PL/SQL script that traps an invalid datatype value given and displays a custom error message

**Procedures**

1. Create a stored procedure that displays the name and salary of the employee whose employee id is passed as parameter
2. Write a stored procedure namely Raise that receives the employee number and raise-in-salary as parameter. It then raises the salary of that employee in the employee table
3. Create a stored procedure that displays the number of employees from table employee whose joined after a given date or joined within a specified period whose start and end dates are provided.

**Triggers**

1. Create a trigger that displays the number of employees after every delete in the employee table
2. Create a trigger that inserts the old row in the Achieve table along with current date, whenever a row is changed in employee table
3. An HR system has an employee table that holds a row for each employee within the company. Each record in the table has a manager field (mgr) that holds the id for the employees’ manager. Write a trigger so that when a manager record is deleted, the mgr field of that manager’s employee is set to NULL
4. Create a trigger to fill in the ID field of students with a value generated from the student\_sequence sequence every time insertion or updation takes place in Student table
5. Write a database trigger that allows changes to employees table only during the business hours (ie from 8 am to 5 pm) from Monday to Saturday. There is no restriction from viewing data from the table
6. Create a trigger which will be implicitly activated based on an update done to the table called new\_donation. After each update, it should collect the sum of the contributions and put it into table total\_donations

SELECT DeptName, MAX(Salary) FROM Employee e RIGHT JOIN Department d ON e.DeptId = d.DeptID GROUP BY DeptName;

1. Read more: <http://java67.blogspot.com/2013/04/10-frequently-asked-sql-query-interview-questions-answers-database.html#ixzz45mM9M4i6>