**DBMSLab-Assignment 2**

Q1: Create a new user making “your\_name” as user-name and “your\_surname” as the password.

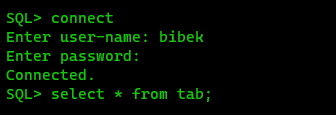
🡺create user bibek identified by root;

Q2: Grant all privileges to the newly created user.

🡺 grant all privileges to bibek;

Q3: connect to the new user.

🡺connect



Q4: Create a table employee with attributes emp\_id, f\_name , l\_name , job\_type, salary, commision, dept, and manager\_id.

🡺 CREATE TABLE Employee (

employee\_id INT,

first\_name VARCHAR(10),

last\_name VARCHAR(10),

job\_types varchar(10),

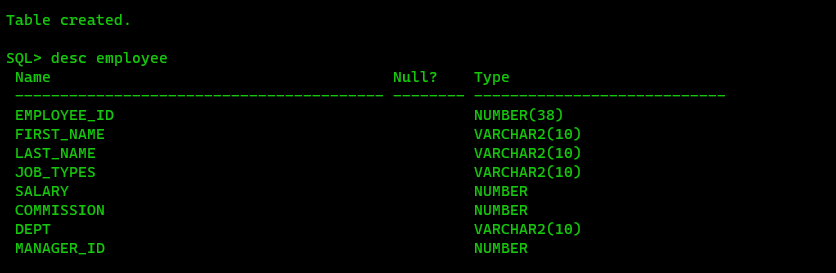
salary number,

commission number,

dept VARCHAR(10),

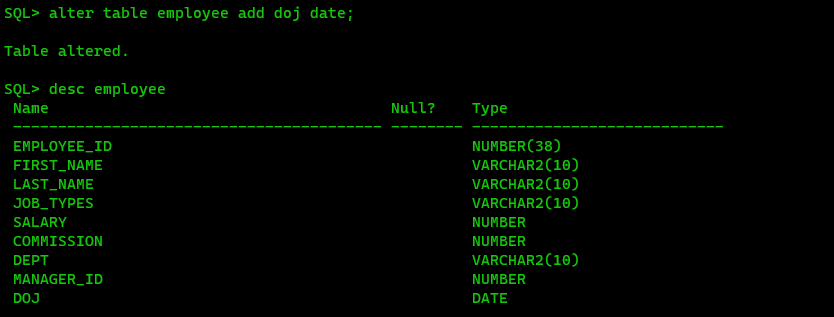
manager\_id number);

Q5: Describe the table employee

🡺 DESC employee

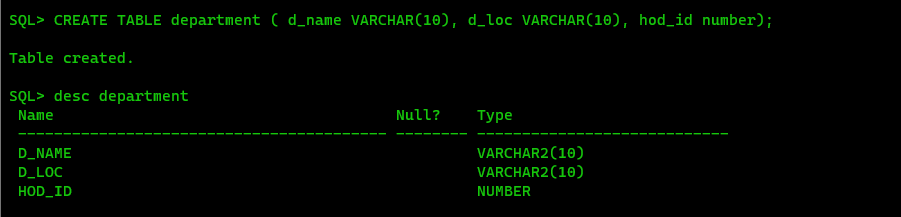
Q6: Add a new column doj to the employee table.

🡺alter table employee add doj date;



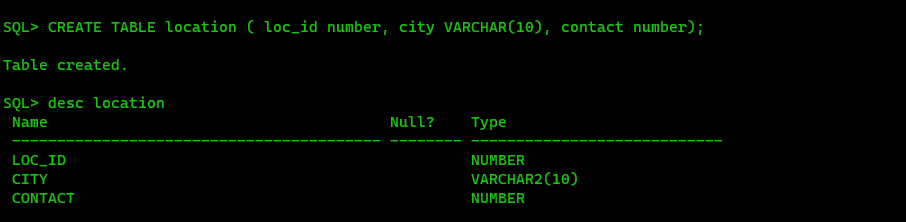
Q7: Create a new table department with attributes d\_name, d\_loc, and hod\_id.

🡺 CREATE TABLE department ( d\_name VARCHAR(10), d\_loc VARCHAR(10), hod\_id number);



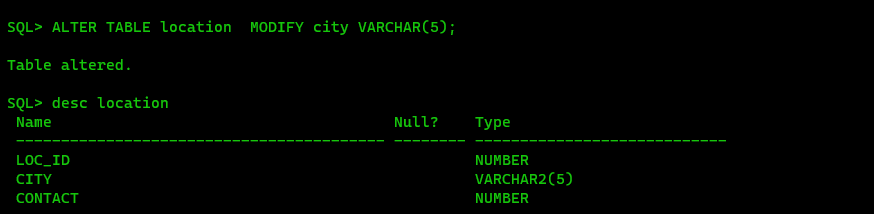
Q8: Create another table named location with attributes loc\_id, city and contact\_no.

🡺 CREATE TABLE location ( loc\_id number, city VARCHAR(10), contact number);



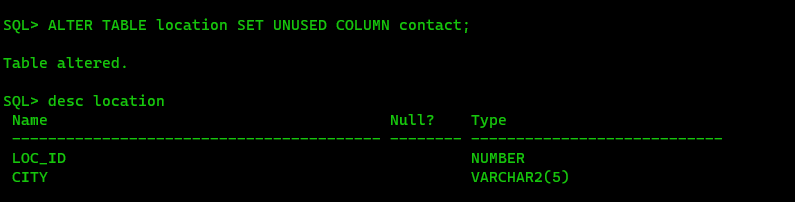
Q9. Enhance the size of city attribute in location table by 5.

🡺 ALTER TABLE location MODIFY city VARCHAR(5);



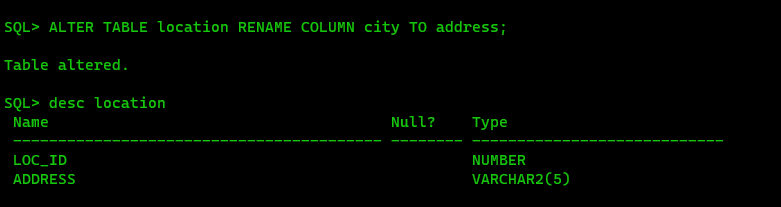
Q10. Delete the contact\_no attribute in the location table.

🡺 ALTER TABLE location SET UNUSED COLUMN contact;



Q11. Rename the city attribute in the location table to address.

🡺 ALTER TABLE location RENAME COLUMN city TO address;



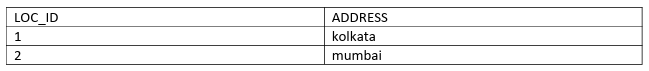
Q12. Change the name of the table from location to loc.

🡺 RENAME location To loc;

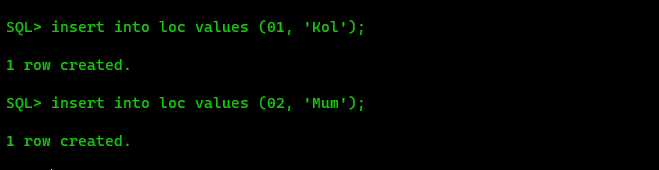
A screen shot of a computer

Description automatically generated

Q13. Insert the following values into the loc table.



🡺 insert into loc values (01, 'Kol');



Q14. Show the values of location table.

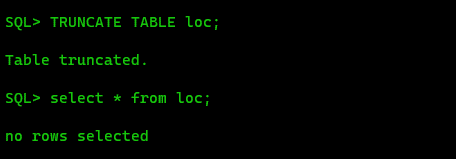
🡺 select \* from loc;

A screenshot of a computer

Description automatically generated

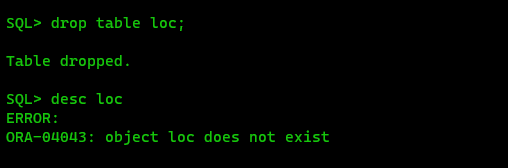
Q15. Delete all values and spaces consumed by loc table.

🡺 TRUNCATE TABLE loc;



Q16. Delete the loc table.

🡺 drop table loc;

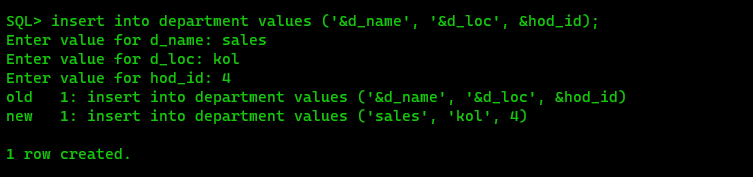


Q17. Insert the following values into the department table.

A blank list of days

Description automatically generated with medium confidence

🡺 insert into department values ('&d\_name', '&d\_loc', &hod\_id);



Q18. Insert the following values into the employee table.

A table of numbers and letters

Description automatically generated with medium confidence

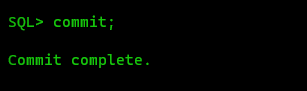
🡺 insert into employee values (&employee\_id, '&first\_name', '&last\_name', '&job\_type', &salary, &commission, '&dept', &manager\_id, '&doj');

A computer screen with green text

Description automatically generated

Q19. Save the database.

🡺commit;



Q20: Show all the attribute values of the department table.

🡺 select \* from department;

A screenshot of a computer program

Description automatically generated

Q21: Display the department names and their locations.

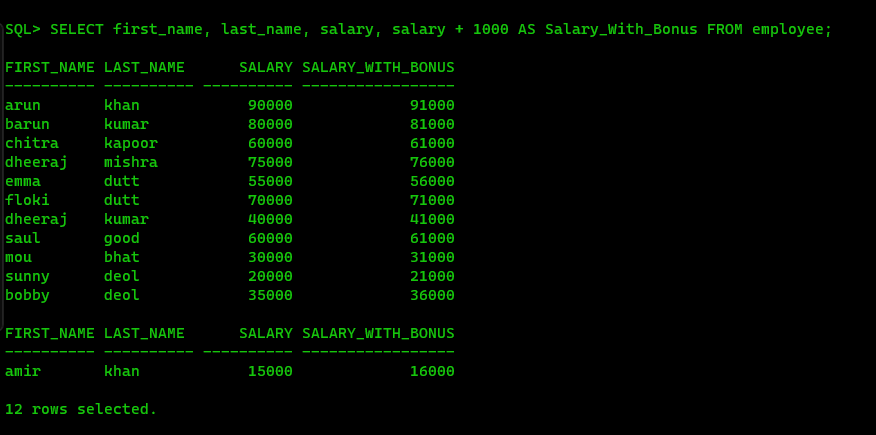
🡺 select d\_name, d\_loc from department;

A screenshot of a computer

Description automatically generated

Q22: Show the employee’s first name, last name, current salary and the salary with a 1000 rupees bonus.

🡺 SELECT first\_name, last\_name, salary, salary + 1000 AS Salary\_With\_Bonus FROM employee;



Q23: Show the employee’s annual salary with a 1000 rupees yearly bonus and the annual salary with a 100 rupees monthly bonus.

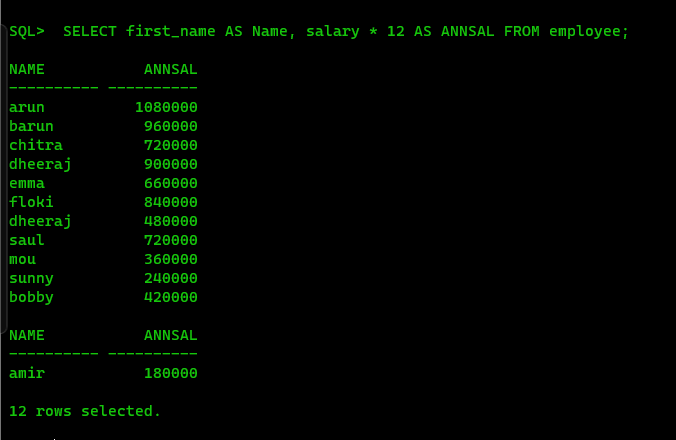
🡺 SELECT first\_name, last\_name, salary \* 12 + 1000 AS Yearly\_Bonus, (salary + 100) \* 12 AS Monthly\_Bonus FROM employee;

A screenshot of a computer screen

Description automatically generated

Q24: Show f\_name as Name and annual salary as ANNSAL from the employee table.

🡺 SELECT first\_name AS Name, salary \* 12 AS ANNSAL FROM employee;



Q25: Show the L\_name as SurName and 100 rupees incremented salary as NewSal from the employee table.

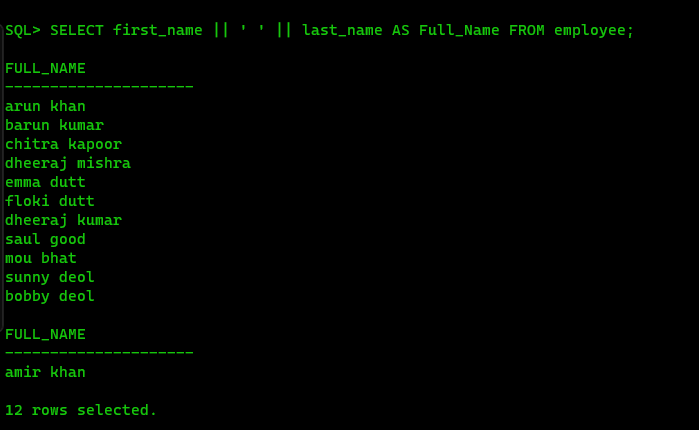
🡺 SELECT last\_name AS SurName, salary + 100 AS NewSal FROM employee;

A screenshot of a computer

Description automatically generated

Q26: Display the employees f\_name and l\_name joined together using the concatenation operator.

🡺 SELECT first\_name || ' ' || last\_name AS Full\_Name FROM employee;



Q27: Show the f\_name, l\_name and job\_type as Employees.

🡺 SELECT first\_name, last\_name, job\_types AS Employees FROM employee;

A screenshot of a computer

Description automatically generated

Q28: Show the employee details in the following fassion:

Employees Details

--------------------------------------------------------

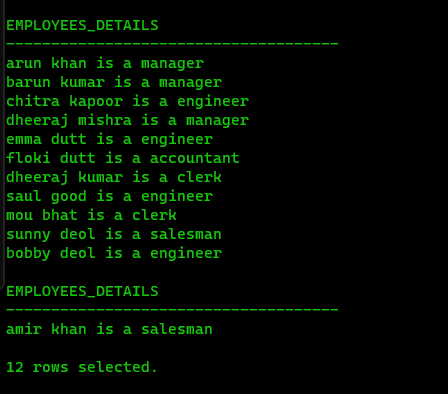
arun khan is a manager

barun kumar is a manager

……..

……..

🡺 SELECT first\_name || ' ' || last\_name || ' is a ' || job\_types AS Employees\_Details FROM employee;



Q29: Show the monthly salary details in the following fassion:

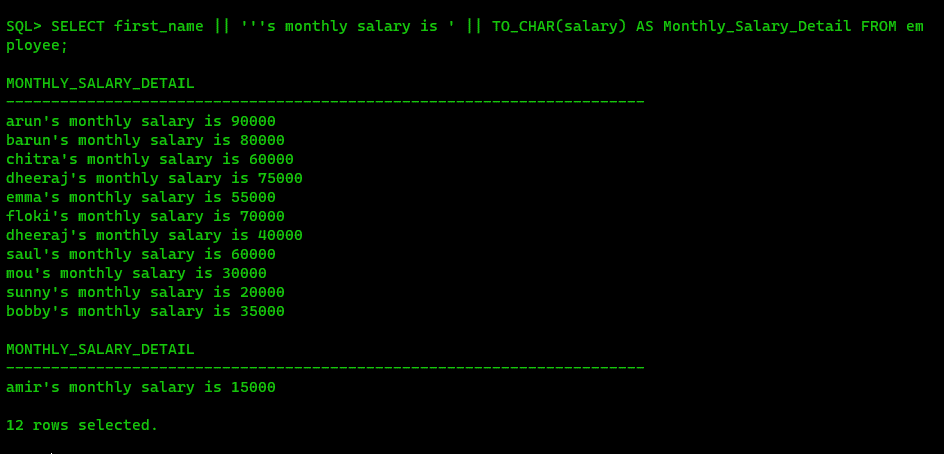
Monthly Salary Details

--------------------------------------------------------------------------------

arun's monthly salary is 90000

………

🡺 SELECT first\_name || '''s monthly salary is ' || TO\_CHAR(salary) AS Monthly\_Salary\_Detalil FROM employee;



Q30: Show the department names from the employee table.

🡺 select dept from employee;

A screenshot of a computer

Description automatically generated

Q31: Show the distinct department names from the employee table.

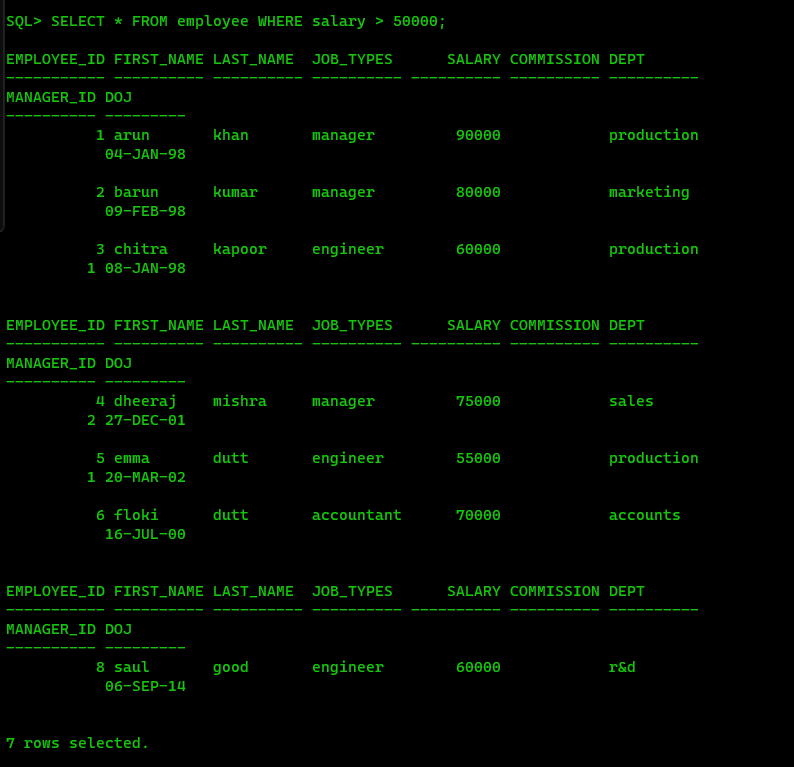
🡺 SELECT DISTINCT dept FROM employee;

A screen shot of a computer

Description automatically generated

Q32: Show the employees earning more than 50000.

🡺 SELECT \* FROM employee WHERE salary > 50000;



Q33. Show the employee’s id’s who are not working under manager id-1.

🡺 SELECT employee\_id FROM employee WHERE manager\_id != 1;

A screen shot of a computer

Description automatically generated

Q34: Show the employee’s names and salaries whose salary ranges between 40000 to 70000.

🡺 SELECT first\_name, last\_name, salary FROM employee WHERE salary BETWEEN 40000 AND 70000;

A black screen with green text

Description automatically generated

Q35: Show the employees who work for manager id 1 or 6 or 8.

🡺 SELECT \* FROM employee WHERE manager\_id IN (1, 6, 8);

A screenshot of a computer screen

Description automatically generated

Q36: Select the first names and salaries of those employee whose last name is khan.

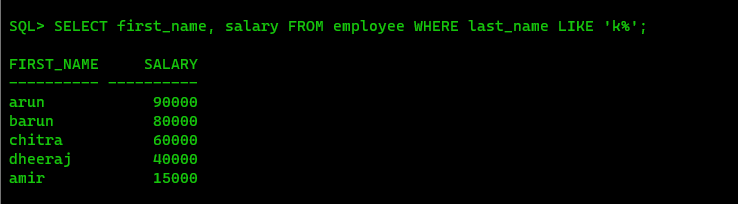
🡺 SELECT first\_name, salary FROM employee WHERE last\_name = 'khan';

A black screen with green text

Description automatically generated

Q37: Select the first names and salaries of those employee whose last name starts with k.

🡺 SELECT first\_name, salary FROM employee WHERE last\_name LIKE 'k%';



Q38: Select the first name, last name and salary of those employee whose last name starts with k and ends with r.

🡺 SELECT first\_name, last\_name, salary FROM employee WHERE last\_name LIKE 'k%r';

A black screen with green text

Description automatically generated

Q39: Select the employees whose 3rd letter of their last name is o.

🡺 SELECT \* FROM employee WHERE SUBSTR(last\_name, 3, 1) = 'o';

A screenshot of a computer

Description automatically generated

Q40: Select the employees who are not working under any manager.

🡺 SELECT \* FROM employee WHERE manager\_id IS NULL;

A screenshot of a computer screen

Description automatically generated

Q41: Select the employees who work as engineers with salary greater than 50000.

🡺 SELECT \* FROM employee WHERE job\_types = 'engineer' AND salary > 50000;

A screenshot of a computer

Description automatically generated

Q42: Select the employees who work in the production department or earns more than 60000.

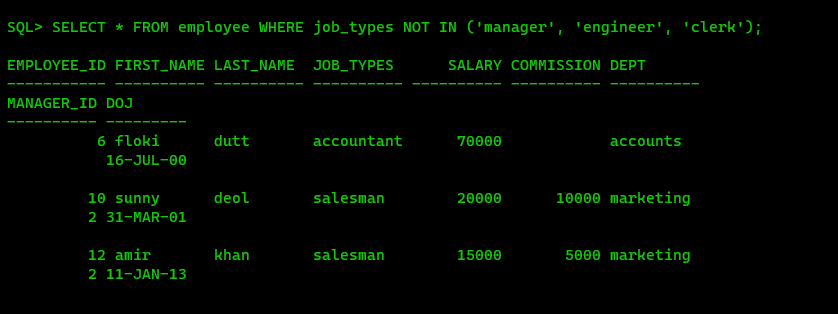
🡺 SELECT \* FROM employee WHERE dept = 'production' OR salary > 60000;

A screenshot of a computer screen

Description automatically generated

Q43: Select those employees who are not managers or engineers or clerks.

🡺 SELECT \* FROM employee WHERE job\_types NOT IN ('manager', 'engineer', 'clerk');



Q44: Select the employees who earns more than 49000 or less than 29000.

🡺 SELECT \* FROM employee WHERE salary > 49000 OR salary < 29000;

A screenshot of a computer

Description automatically generated

Q45. Select the employees who don’t have an ‘o’ as the 2nd last letter of their last name.

🡺 SELECT \* FROM employee WHERE SUBSTR(last\_name, -2, 1) != 'o';

A screenshot of a computer

Description automatically generated

Q46. Select the employees who get commission.

🡺 SELECT \* FROM employee WHERE commission IS NOT NULL;

A screenshot of a computer

Description automatically generated

Q47. WAQ to display the current date.

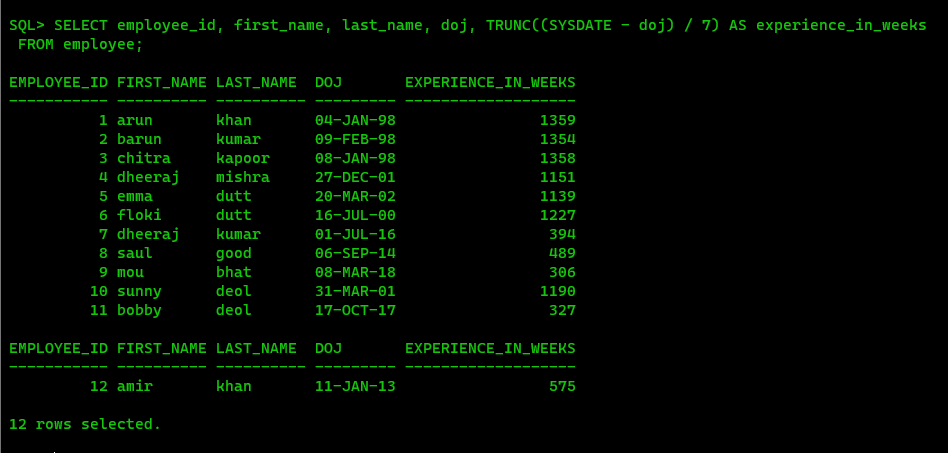
🡺 SELECT SYSDATE AS current\_date FROM dual;

A black screen with green text

Description automatically generated

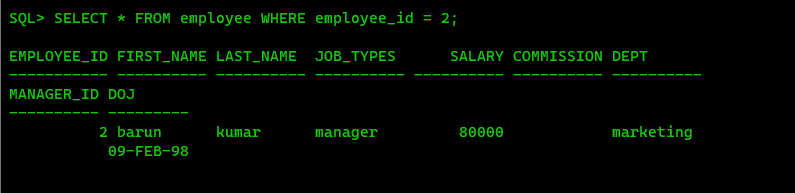
Q48. Show the total experience in weeks for all the employees.

🡺 SELECT employee\_id, first\_name, last\_name, doj, TRUNC((SYSDATE - doj) / 7) AS experience\_in\_weeks FROM employee;



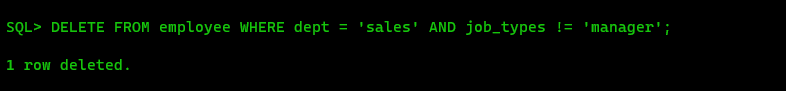
Q49. Find the employees working under employee\_id 2.

🡺 SELECT \* FROM employee WHERE employee\_id = 2;



Q50. Delete the employees from sales department if they are not working as managers.

🡺 DELETE FROM employee WHERE dept = 'sales' AND job\_types != 'manager';

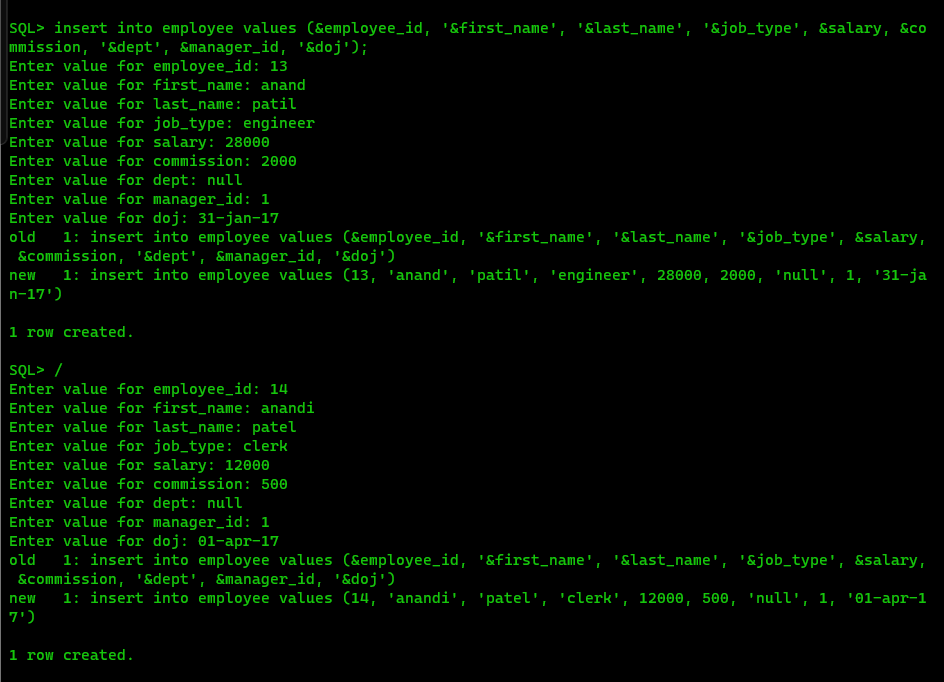


Q51. Insert the following two rows in the employee table without inserting any value in the department field.

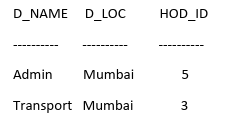
A number of numbers on a white background

Description automatically generated

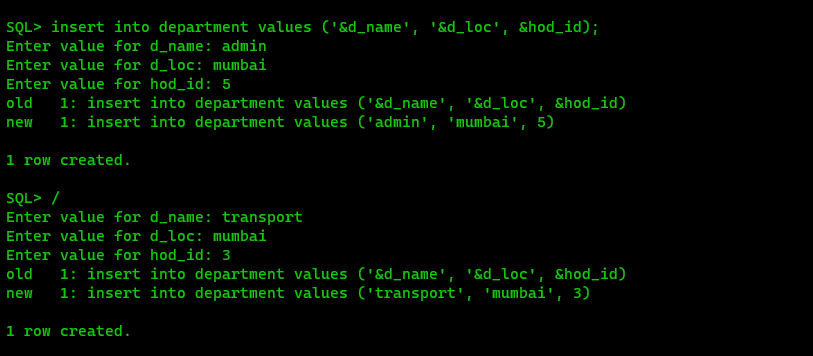
🡺 insert into employee values (&employee\_id, '&first\_name', '&last\_name', '&job\_type', &salary, &commission, '&dept', &manager\_id, '&doj');



Q52. . Insert the following two rows in the department table.

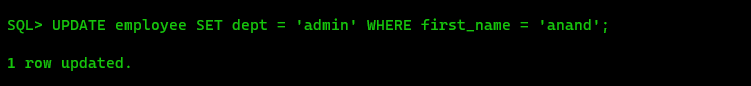


🡺 insert into department values ('&d\_name', '&d\_loc', &hod\_id);



Q53. Update the employee table. Assign Anand to the admin department.

🡺 UPDATE employee SET dept = 'admin' WHERE first\_name = 'anand';



Q54. Update the manager\_id from 2 to 1 in the employee table.

🡺 UPDATE employee SET manager\_id = 1 WHERE manager\_id = 2;

A black screen with green text

Description automatically generated

Q55. Display the employee details in descending order on their salary.

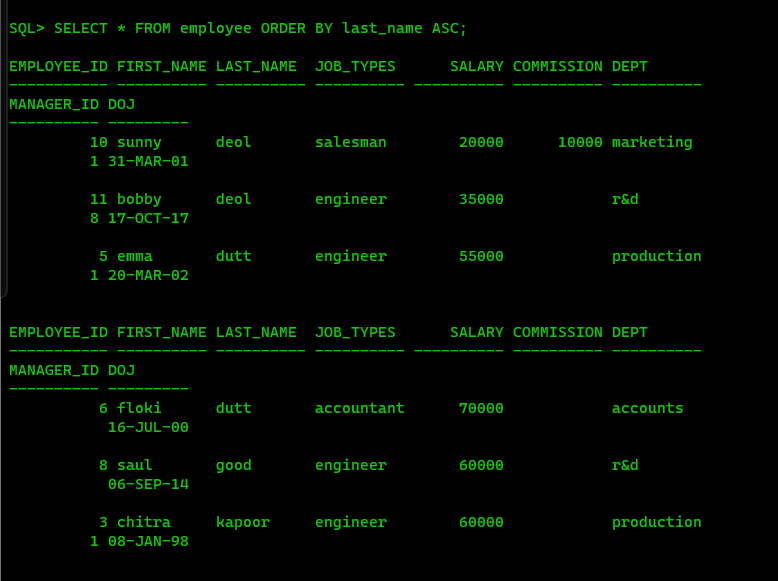
🡺 SELECT \* FROM employee ORDER BY salary DESC;

A screenshot of a computer

Description automatically generated

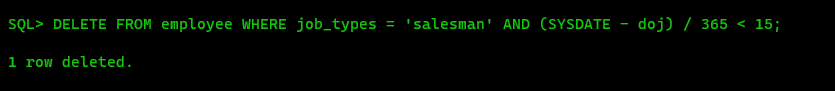
Q56. Display the employee details in ascending order on their l\_name.

🡺 SELECT \* FROM employee ORDER BY last\_name ASC;



Q57. Delete the employees who are working as salesmen and having less experience than 15 years.

🡺 DELETE FROM employee WHERE job\_types = 'salesman' AND (SYSDATE - doj) / 365 < 15;



Q58. Commit the database.

🡺commit;

