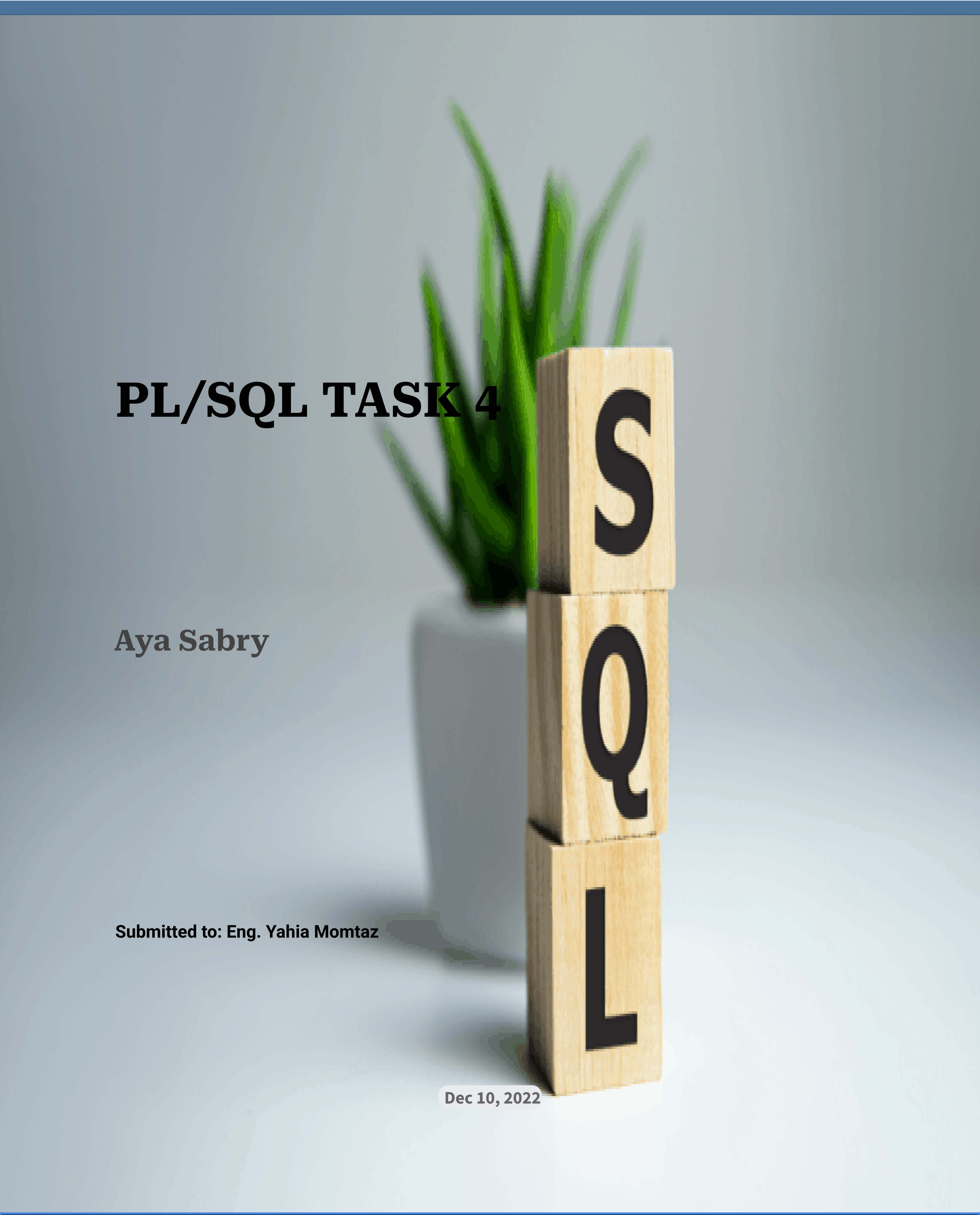


PL/SQL TASK 4

Aya Sabry

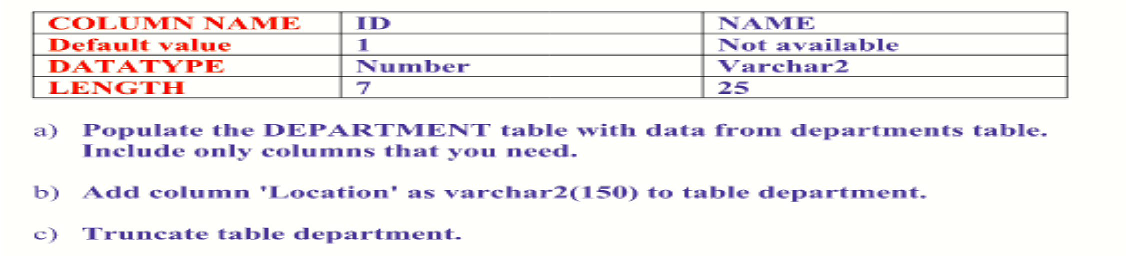
Submitted to: Eng. Yahia Momtaz

Dec 10, 2022

Task 4

1. Create the DEPARTMENT table based on the following table instance chart.

Confirm that the table is created.



COLUMN NAME	ID	NAME
Default value	1	Not available
DATATYPE	Number	Varchar2
LENGTH	7	25

a) Populate the DEPARTMENT table with data from departments table. Include only columns that you need.

b) Add column 'Location' as varchar2(150) to table department.

c) Truncate table department.

CREATE TABLE department

(id NUMBER(7) default 1,   name VARCHAR2(25));

select * from department;

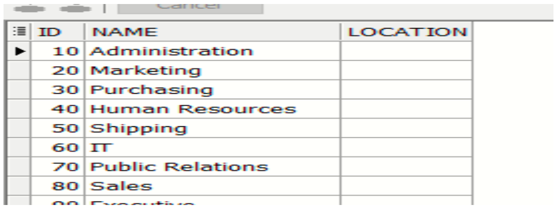
a) INSERT INTO department

SELECT department_id, department_name

FROM departments;

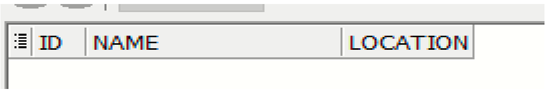
b) alter table department

add location varchar2(150);



ID	NAME	LOCATION
10	Administration	
20	Marketing	
30	Purchasing	
40	Human Resources	
50	Shipping	
60	IT	
70	Public Relations	
80	Sales	
90	Executive	

c)truncate table department;



ID	NAME	LOCATION

2. Create table employee based on the structure of the employees

table(Structure with data).

Include only the employee_id, last_name, email ,salary and

department_id columns

Employee_id Primary key

email unique

```
  CREATE TABLE employee as
  SELECT employee_id  , last_name, email,salary,
  department_id
  FROM employees;
  select *from employee;
  ALTER table employee
  MODIFY  employee_id NUmber(6) NOT NULL;
  ALTER table employee
ADD CONSTRAINT PK_id PRIMARY KEY (employee_id);
 ALTER table employee
 add constraint email_uniq unique( email );
```

3. Create the following tables using ddl

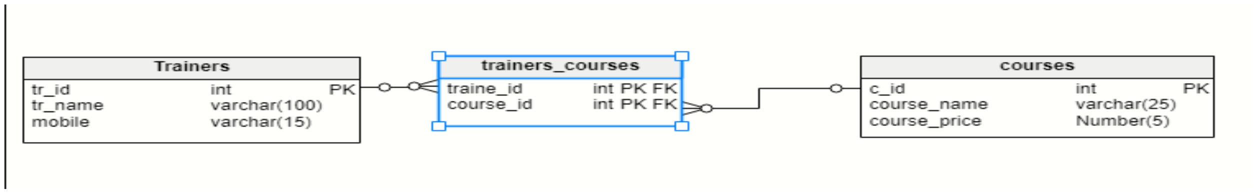
Trainers [tr_id, tr_name, tr_mobile]

Courses [crs_id, crs_name, crs_price]

Use Many to Many relationship;

Solve using create tables, then alter trainers and add email column then alter

again to add unique constraints;



se insert to set those data

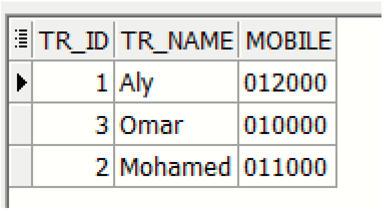
Trainer [aly] &gt; teach [php – oracle – java]

Trainer [Mohamed] &gt; teach [oracle]

Trainer [Omar] &gt; teach [oracle – java]

Then select the data using inner join

```
CREATE TABLE courses(

   c_id Number(3)  NOT NULL,

   course_name varchar2(25)  NOT NULL,

  course_price Number(3)  NOT NULL,

   CONSTRAINT course_pk PRIMARY KEY (c_id));

CREATE TABLE  trainers_courses( tr_id Number(3)  NOT NULL,

   c_id  Number(3)  NOT NULL,

   CONSTRAINT trainer_course_pk PRIMARY KEY (tr_id,c_id));

CREATE TABLE Trainers ( tr_id Number(3)  NOT NULL,

   tr_name varchar2(100)  NOT NULL,

   mobile varchar2(15)  NOT NULL,

   CONSTRAINT trainer_pk PRIMARY KEY (tr_id));

INSERT INTO  Trainers (TR_ID, TR_NAME, MOBILE) VALUES (1, 'Aly', '012000');

INSERT INTO  Trainers (TR_ID, TR_NAME, MOBILE) VALUES (2, 'Mohamed', '011000');

INSERT INTO  Trainers (TR_ID, TR_NAME, MOBILE) VALUES (3, 'Omar', '010000');
```

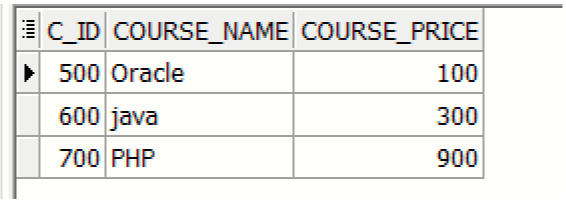


TR_ID	TR_NAME	MOBILE
1	Aly	012000
3	Omar	010000
2	Mohamed	011000

INSERT INTO courses (C_ID, COURSE_NAME, COURSE_PRICE) VALUES (500, 'Oracle', 100);

INSERT INTO courses (C_ID, COURSE_NAME, COURSE_PRICE) VALUES (600, 'java', 300);

INSERT INTO courses (C_ID, COURSE_NAME, COURSE_PRICE) VALUES (700, 'PHP', 900);



C_ID	COURSE_NAME	COURSE_PRICE
500	Oracle	100
600	java	300
700	PHP	900

INSERT INTO trainers_courses (TR_ID, C_ID) VALUES (1,500);

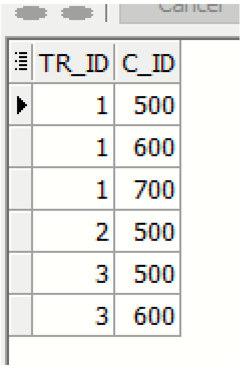
INSERT INTO trainers_courses (TR_ID, C_ID) VALUES (1,600);

INSERT INTO trainers_courses (TR_ID, C_ID) VALUES (1,700);

INSERT INTO trainers_courses (TR_ID, C_ID) VALUES (2,500);

INSERT INTO trainers_courses (TR_ID, C_ID) VALUES (3,500);

INSERT INTO trainers_courses (TR_ID, C_ID) VALUES (3,600);

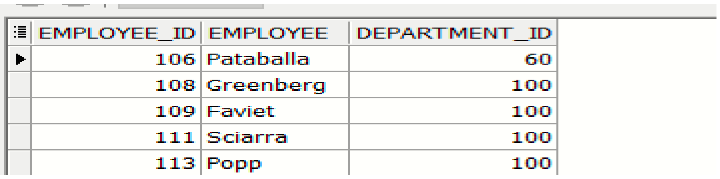


TR_ID	C_ID
1	500
1	600
1	700
2	500
3	500
3	600

4. Create a view called EMP_VU based on the employee number, employee name, and department number from the EMPlOYEES table. Change the heading for the employee name to EMPLOYEE

create or replace VIEW EMP_VU AS

SELECT employee_id, last_name employee, department_id FROM employees;

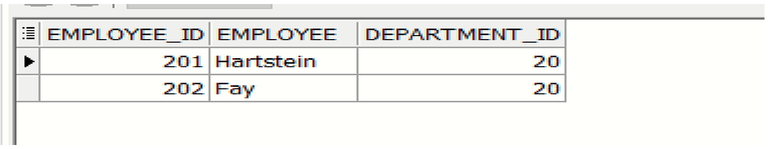


EMPLOYEE_ID	EMPLOYEE	DEPARTMENT_ID
106	Pataballa	60
108	Greenberg	100
109	Faviet	100
111	Sciarra	100
113	Popp	100

Modify the EMP_VU view to display the employees in department 20. Note:

the view can't be used to manipulate the employees in departments other than 20.

create or replace VIEW EMP_VU AS SELECT employee_id, last_name employee, department_id FROM employees WHERE department_id= 20 WITH CHECK OPTION;



EMPLOYEE_ID	EMPLOYEE	DEPARTMENT_ID
201	Hartstein	20
202	Fay	20

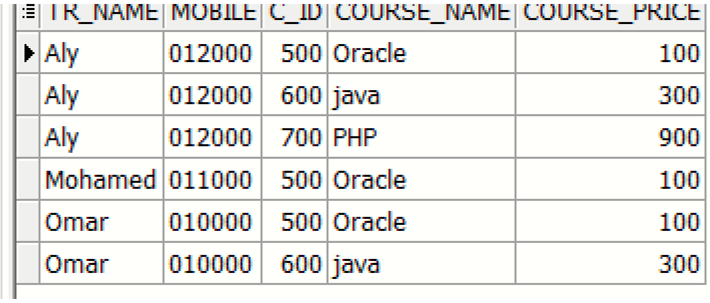
select t.tr_name,t.mobile ,cr.c_id,cr.course_name, cr.COURSE_PRICE

from trainers t

inner join trainers_courses tc

on t.tr_id=tc.tr_id inner join courses cr

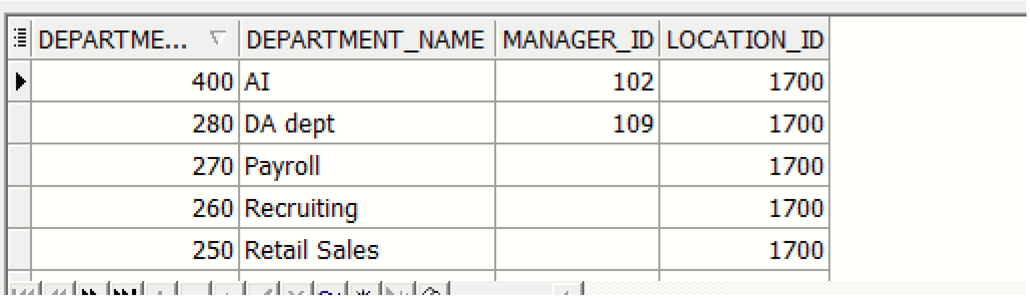
on tc.c_id =cr.c_id;



TR_NAME	MOBILE	C_ID	COURSE_NAME	COURSE_PRICE
Aly	012000	500	Oracle	100
Aly	012000	600	java	300
Aly	012000	700	PHP	900
Mohamed	011000	500	Oracle	100
Omar	010000	500	Oracle	100
Omar	010000	600	java	300

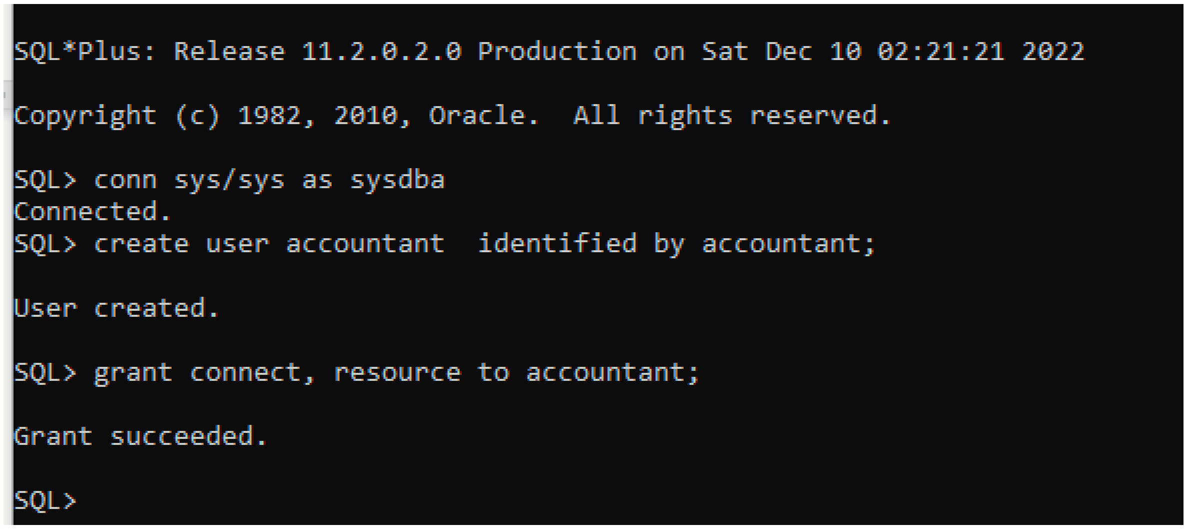
5. Create a sequence to be used with the primary key column of the DEPARTMENTS table. The sequence should start at 400 and have a maximum value of 1000. Have your sequence increment by ten numbers. Name the sequence DEPT_ID_SEQ. and use it to insert a new row in departments table

```
  CREATE SEQUENCE DEPT_ID_SEQ
  START WITH 400
  INCREMENT BY 10
  MAXVALUE 1000;
insert into departments
(DEPARTMENT_ID, DEPARTMENT_NAME, MANAGER_ID, LOCATION_ID )
values
(DEPT_ID_SEQ.nextval , 'AI', 102, 1700);
```



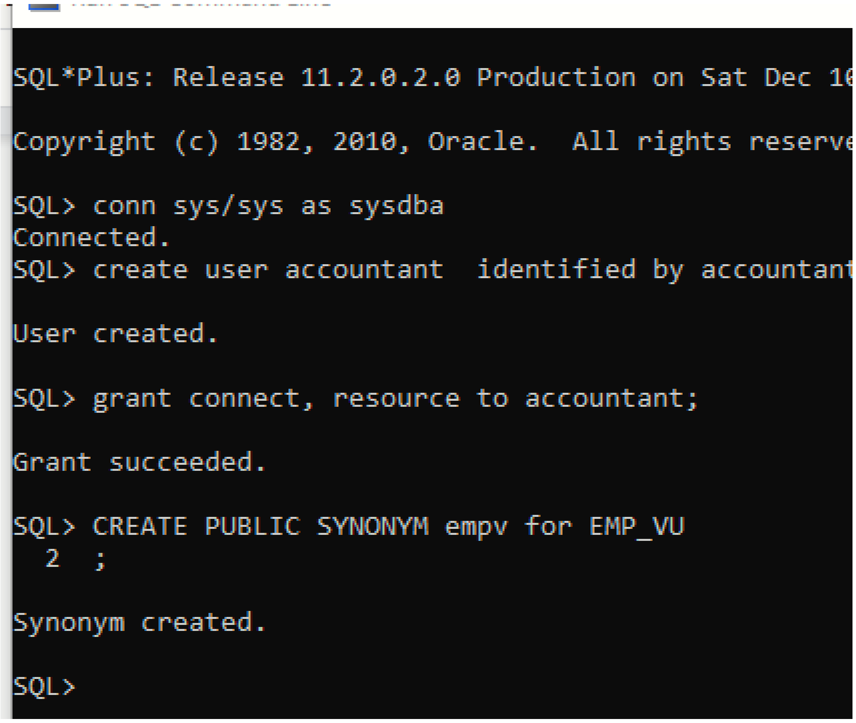
DEPARTME...	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
400	AI	102	1700
280	DA dept	109	1700
270	Payroll		1700
260	Recruiting		1700
250	Retail Sales		1700

6. Create new user “accountant” grant this user two system roles with minimum privileges to access the system.



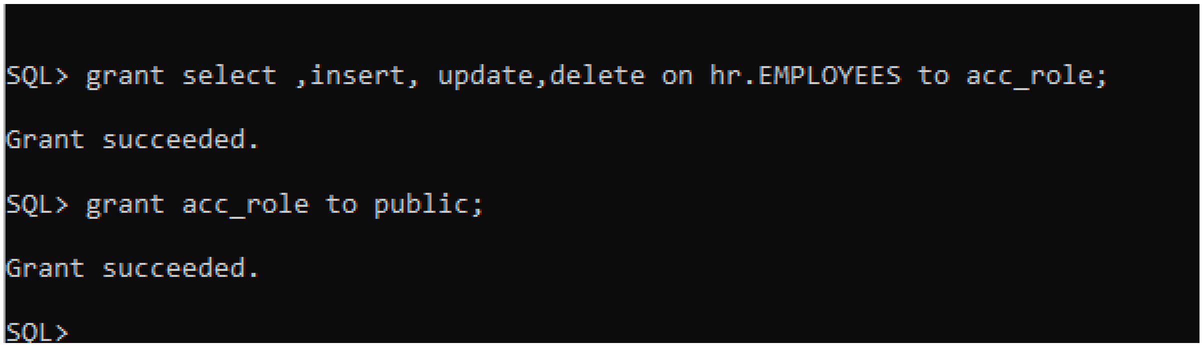
```
SQL*Plus: Release 11.2.0.2.0 Production on Sat Dec 10 02:21:21 2022

Copyright (c) 1982, 2010, Oracle.  All rights reserved.

SQL> conn sys/sys as sysdba
Connected.
SQL> create user accountant  identified by accountant;

User created.

SQL> grant connect, resource to accountant;

Grant succeeded.

SQL>
```

7. Create public synonyms for the view EMP_VU.



```
SQL*Plus: Release 11.2.0.2.0 Production on Sat Dec 1

Copyright (c) 1982, 2010, Oracle.  All rights reserve

SQL> conn sys/sys as sysdba
Connected.
SQL> create user accountant  identified by accountant

User created.

SQL> grant connect, resource to accountant;

Grant succeeded.

SQL> CREATE PUBLIC SYNONYM empv for EMP_VU
  2  ;

Synonym created.

SQL>
```

8. Create role to select and do DML operations on the EMPLOYEES table in your schema, grant access on this role to all users.



```
SQL> grant select ,insert, update,delete on hr.EMPLOYEES to acc_role;

Grant succeeded.

SQL> grant acc_role to public;

Grant succeeded.

SQL>
```