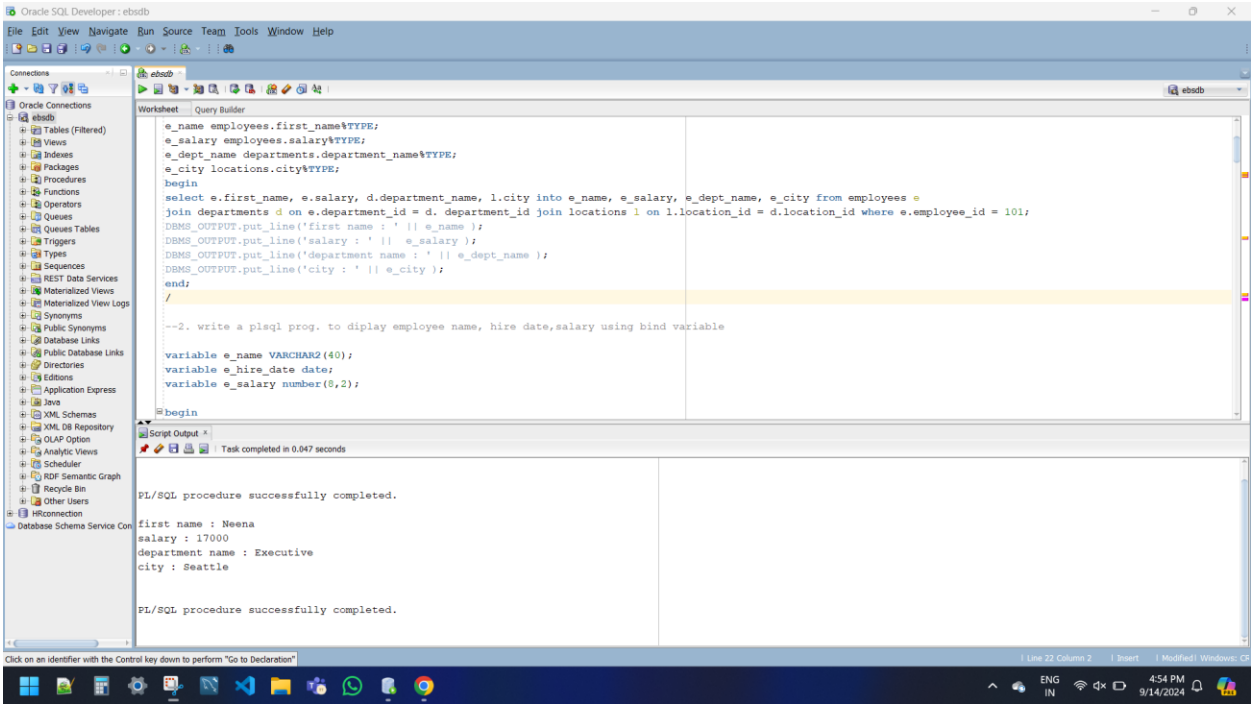
```
set serveroutput on;


--1. write a plsql program to display employee first name,salary, dept name, city

set SERVEROUTPUT ON;

--select * from employees;

--select * from departments;

--select * from locations;


declare

e_name employees.first_name%TYPE;

e_salary employees.salary%TYPE;

e_dept_name departments.department_name%TYPE;

e_city locations.city%TYPE;

begin

select e.first_name, e.salary, d.department_name, l.city into e_name, e_salary,
e_dept_name, e_city from employees e

join departments d on e.department_id = d. department_id join locations l on l.location_id
= d.location_id where e.employee_id = 101;

DBMS_OUTPUT.put_line('first name : ' || e_name );

DBMS_OUTPUT.put_line('salary : ' ||  e_salary );

DBMS_OUTPUT.put_line('department name : ' || e_dept_name );

DBMS_OUTPUT.put_line('city : ' || e_city );

end;

/
```



--2. write a plsql prog. to diplay employee name, hire date,salary using bind variable

set autoprint on;

variable e_name VARCHAR2(40);

variable e_hire_date date;

variable e_salary number(8,2);

begin

select first_name, hire_date, salary into :e_name, :e_hire_Date, :e_salary from employees where employee_id = 101;

DBMS_OUTPUT.put_line('first name : ' || :e_name);

DBMS_OUTPUT.put_line('hire date : ' || :e_hire_Date);

DBMS_OUTPUT.put_line('salary : ' || :e_salary);

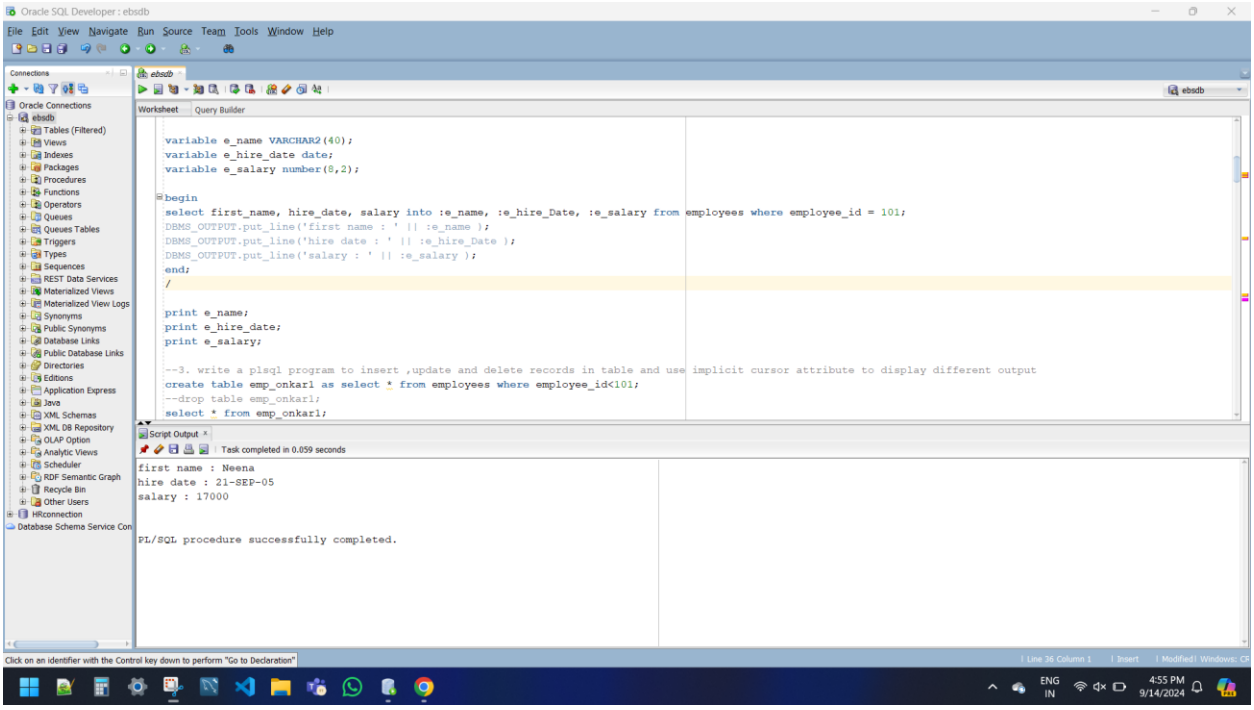
end;

/

print e_name;

print e_hire_date;

print e_salary;



--3. write a plsql program to insert ,update and delete records in table and use implicit cursor attribute to display different output

create table emp_onkar1 as select * from employees where employee_id<101;

--drop table emp_onkar1;

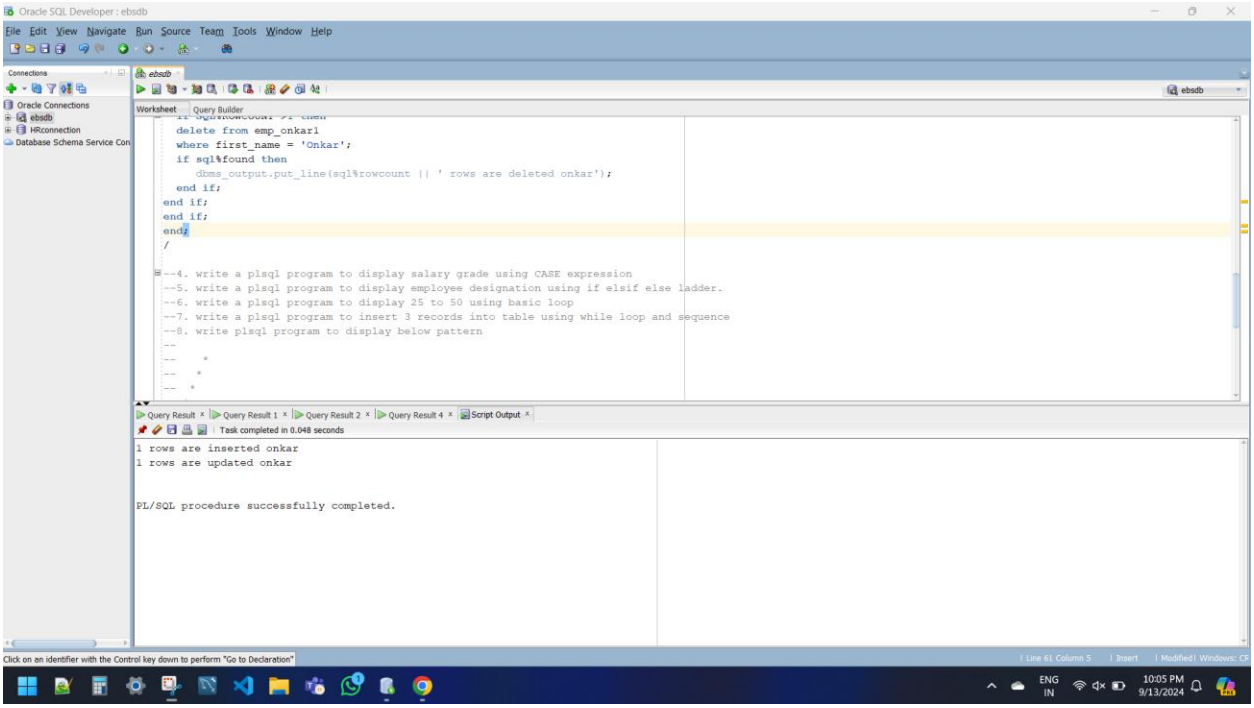
select * from emp_onkar1;

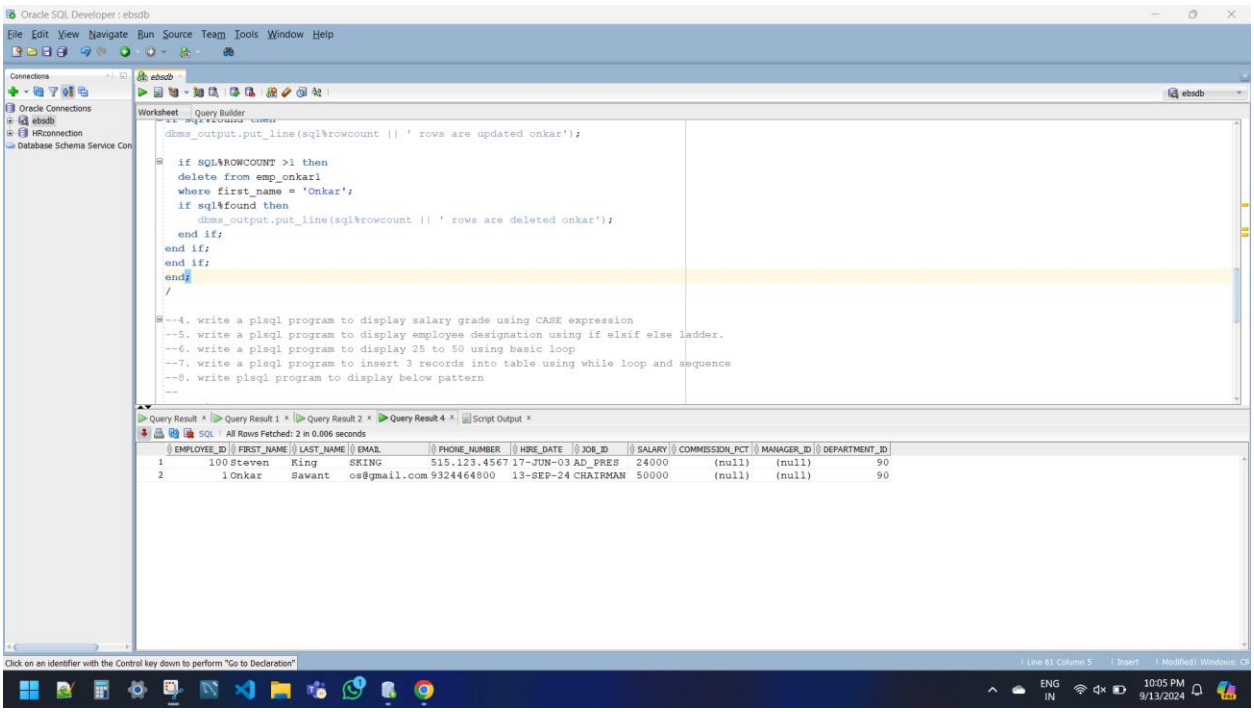
begin

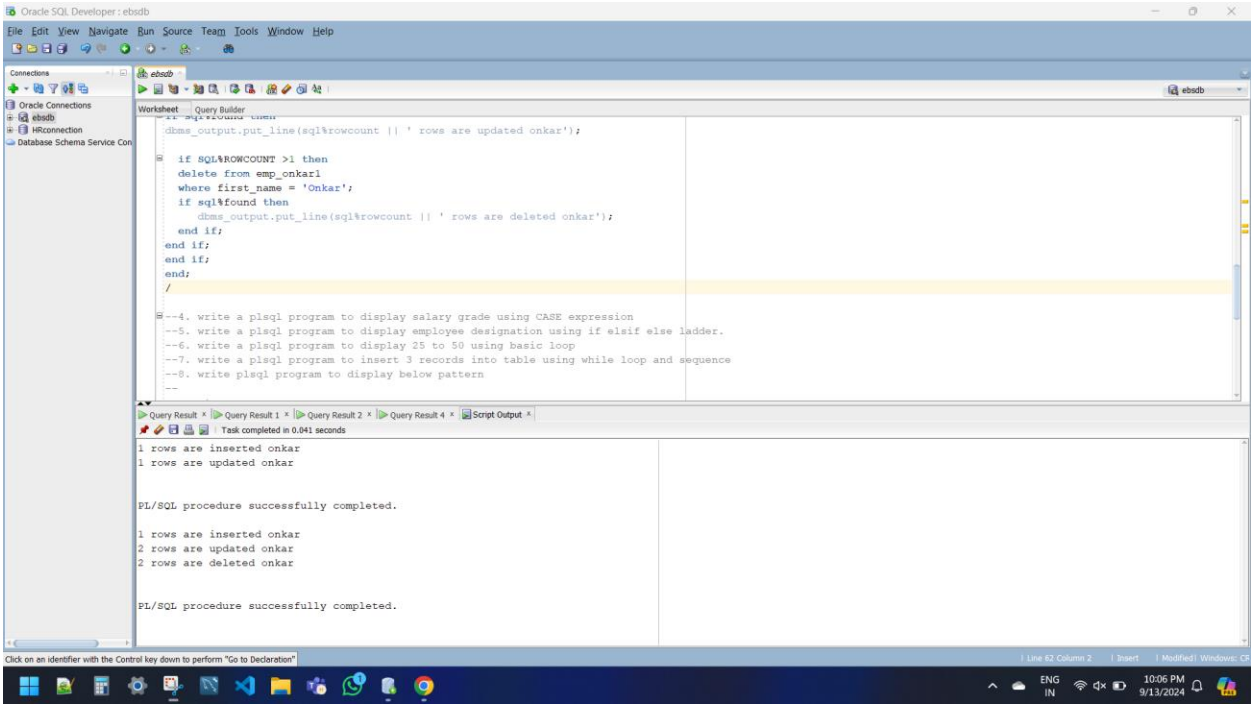
insert into emp_onkar1
values(1,'Onkar','Sawant','os@gmail.com','9324464800',sysdate,'CHAIRMAN',30000,null,
null,90);

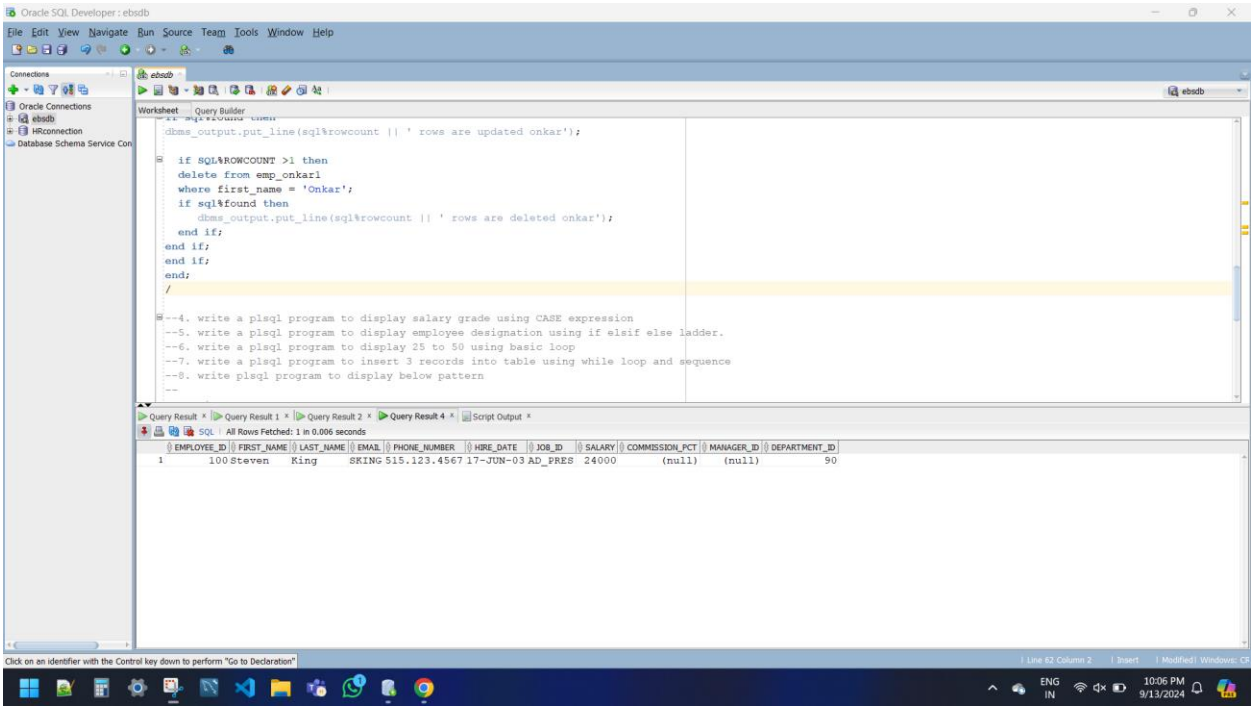
if sql%found then

```
dbms_output.put_line(sql%rowcount || ' rows are inserted onkar');
end if;

update emp_onkar1
set salary = 50000
where first_name = 'Onkar';
if sql%found then
dbms_output.put_line(sql%rowcount || ' rows are updated onkar');

 if SQL%ROWCOUNT >1 then
 delete from emp_onkar1
 where first_name = 'Onkar';
 if sql%found then
   dbms_output.put_line(sql%rowcount || ' rows are deleted onkar');
 end if;
end if;
end if;
end;
/
```


Oracle SQL Developer : ebsdb
File Edit View Navigate Run Source Team Tools Window Help
delete from emp_onkar1
where first_name = 'Onkar';
if sql%found then
dbms_output.put_line(sql%rowcount || ' rows are deleted onkar');
end if;
end if;
end if;
end;
/
--4. write a plsql program to display salary grade using CASE expression
--5. write a plsql program to display employee designation using if elsif else ladder.
--6. write a plsql program to display 25 to 50 using basic loop
--7. write a plsql program to insert 3 records into table using while loop and sequence
--8. write plsql program to display below pattern
--
--
--
Query Result
Query Result 1
Query Result 2
Query Result 4
Script Output
Task completed in 0.048 seconds
1 rows are inserted onkar
1 rows are updated onkar
PL/SQL procedure successfully completed.
Click on an identifier with the Control key down to perform "Go to Declaration"
ENG
IN
10:05 PM
9/13/2024


Oracle SQL Developer : ebsdb
File Edit View Navigate Run Source Team Tools Window Help
dbms_output.put_line(sql%rowcount || ' rows are updated onkar');
if SQL%ROWCOUNT >1 then
delete from emp_onkar1
where first_name = 'Onkar';
if sql%found then
dbms_output.put_line(sql%rowcount || ' rows are deleted onkar');
end if;
end if;
end if;
end;
/
--4. write a plsql program to display salary grade using CASE expression
--5. write a plsql program to display employee designation using if elsif else ladder.
--6. write a plsql program to display 25 to 50 using basic loop
--7. write a plsql program to insert 3 records into table using while loop and sequence
--8. write plsql program to display below pattern
--
Query Result
Query Result 1
Query Result 2
Query Result 4
Script Output
SQL | All Rows Fetched: 2 in 0.006 seconds
EMPLOYEE_ID FIRST_NAME LAST_NAME EMAIL PHONE_NUMBER HIRE_DATE JOB_ID SALARY COMMISSION_PCT MANAGER_ID DEPARTMENT_ID
1 100 Steven King SKING 515.123.4567 17-JUN-03 AD_PRES 24000 (null) (null) 90
2 1 Onkar Sawant os@gmail.com 9324464800 13-SEP-24 CHAIRMAN 50000 (null) (null) 90
Click on an identifier with the Control key down to perform "Go to Declaration"
ENG
IN
10:05 PM
9/13/2024



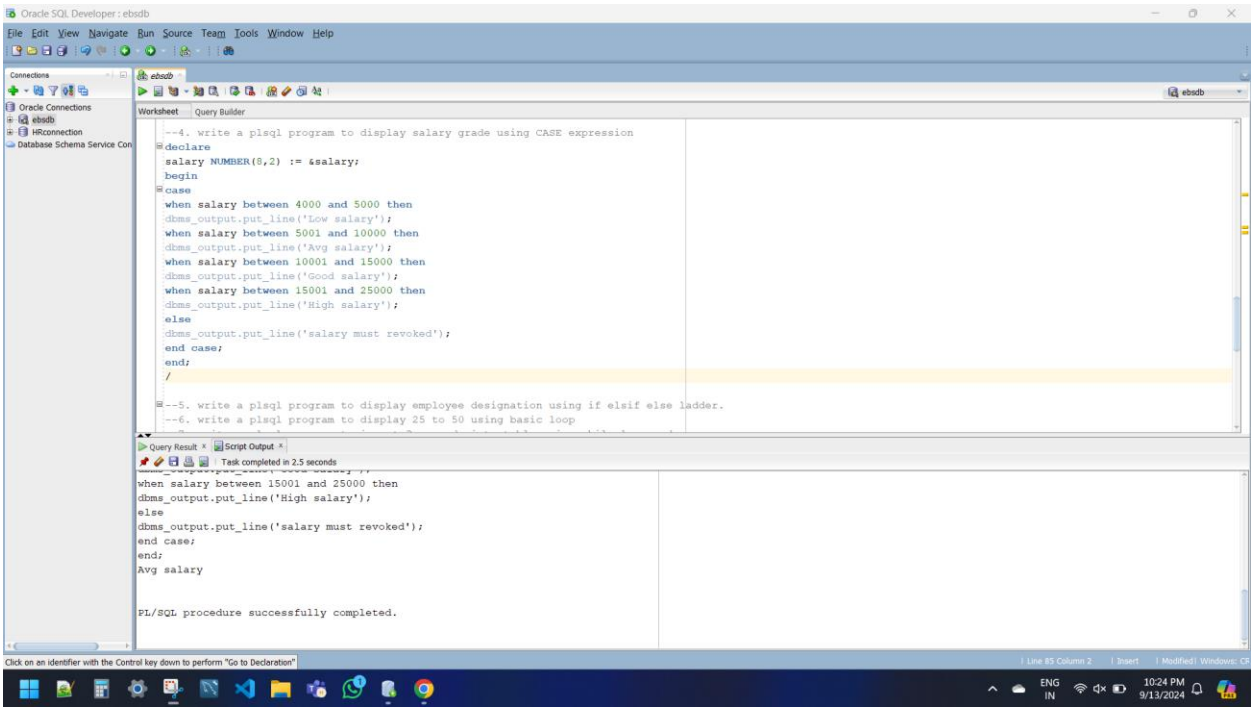


--4. write a plsql program to display salary grade using CASE expression

declare

salary NUMBER(8,2) := &salary;

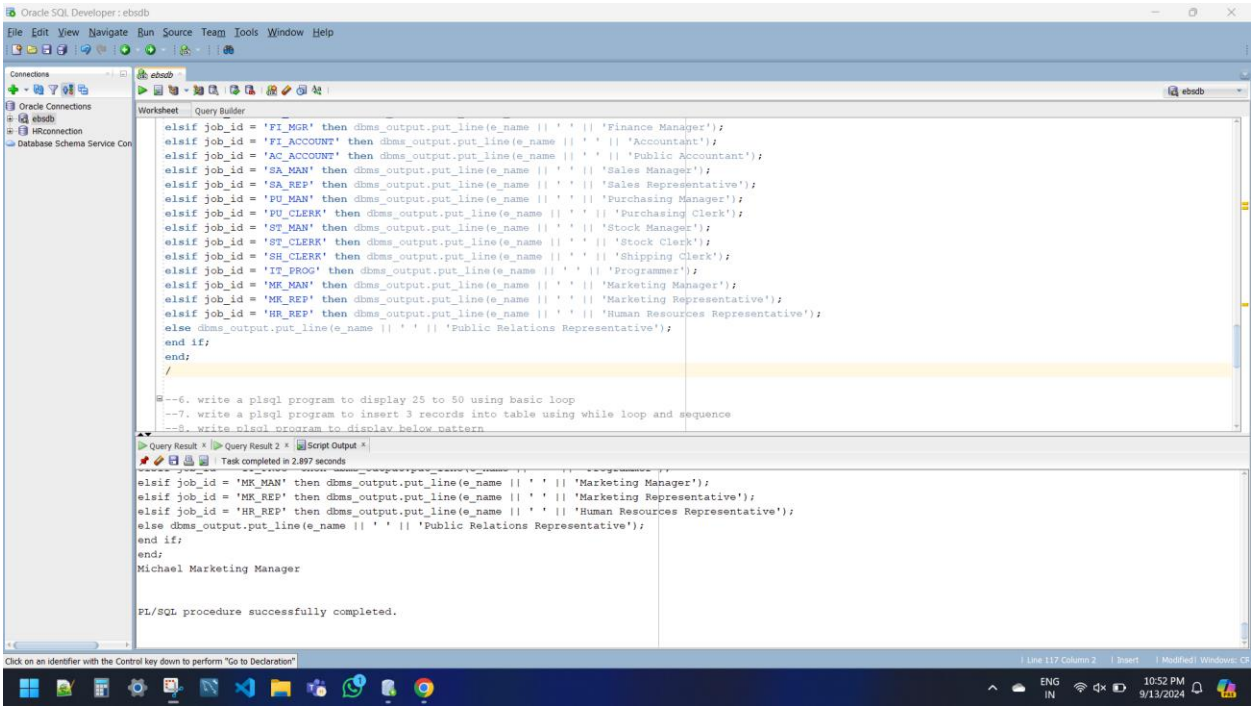
```
begin
case
when salary between 4000 and 5000 then
dbms_output.put_line('Low salary');
when salary between 5001 and 10000 then
dbms_output.put_line('Avg salary');
when salary between 10001 and 15000 then
dbms_output.put_line('Good salary');
when salary between 15001 and 25000 then
dbms_output.put_line('High salary');
else
dbms_output.put_line('salary must revoked');
end case;
end;
/
```



--5. write a plsql program to display employee designation using if elsif else ladder.

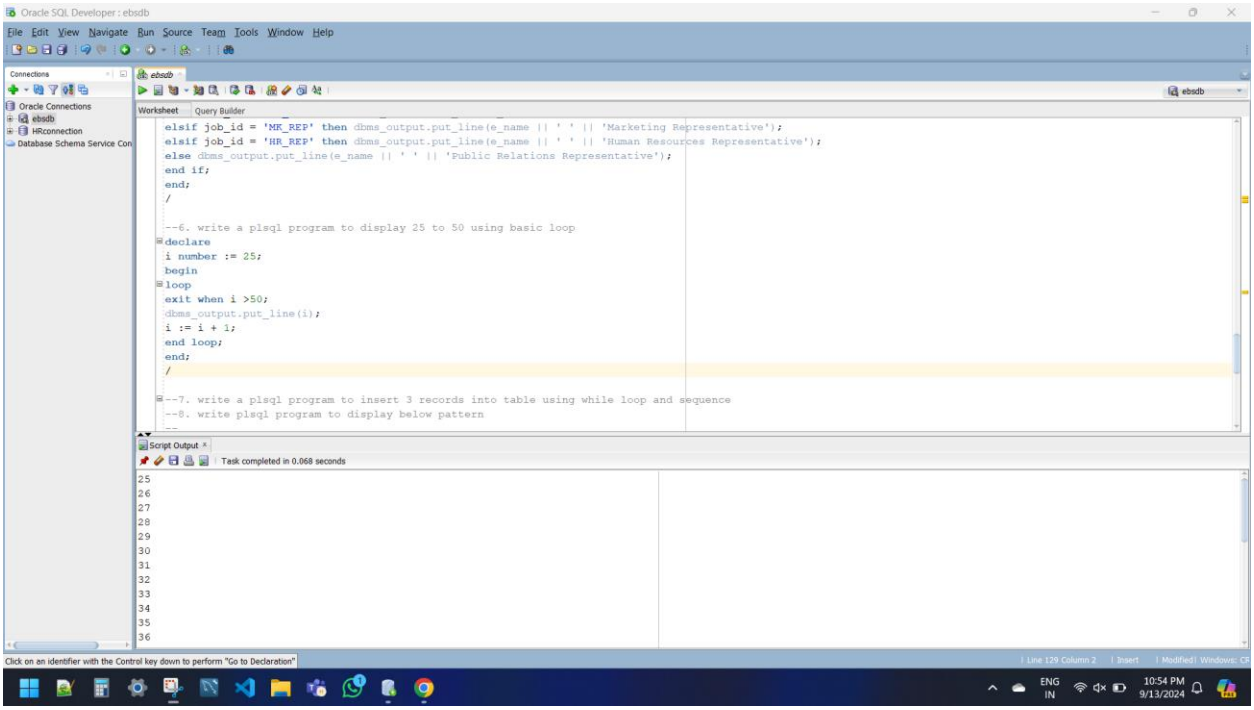
```
select distinct job_id from employees;

select * from jobs;


declare

e_name varchar2(20);

e_id number := &id;

job_id VARCHAR2(20);

begin

select first_name, job_id into e_name, job_id from employees where employee_id = e_id;

if job_id = 'AD_VP' then dbms_output.put_line(e_name || ' ' || 'Administration Vice
President');

elsif job_id = 'AD_PRES' then dbms_output.put_line(e_name || ' ' || 'President');

elsif job_id = 'AD_ASST' then dbms_output.put_line(e_name || ' ' || 'Administration
Assistant');

elsif job_id = 'FI_MGR' then dbms_output.put_line(e_name || ' ' || 'Finance Manager');

elsif job_id = 'FI_ACCOUNT' then dbms_output.put_line(e_name || ' ' || 'Accountant');

elsif job_id = 'AC_ACCOUNT' then dbms_output.put_line(e_name || ' ' || 'Public
Accountant');

elsif job_id = 'SA_MAN' then dbms_output.put_line(e_name || ' ' || 'Sales Manager');

elsif job_id = 'SA_REP' then dbms_output.put_line(e_name || ' ' || 'Sales Representative');

elsif job_id = 'PU_MAN' then dbms_output.put_line(e_name || ' ' || 'Purchasing Manager');

elsif job_id = 'PU_CLERK' then dbms_output.put_line(e_name || ' ' || 'Purchasing Clerk');

elsif job_id = 'ST_MAN' then dbms_output.put_line(e_name || ' ' || 'Stock Manager');

elsif job_id = 'ST_CLERK' then dbms_output.put_line(e_name || ' ' || 'Stock Clerk');

elsif job_id = 'SH_CLERK' then dbms_output.put_line(e_name || ' ' || 'Shipping Clerk');
```

```
elsif job_id = 'IT_PROG' then dbms_output.put_line(e_name || ' ' || 'Programmer');

elsif job_id = 'MK_MAN' then dbms_output.put_line(e_name || ' ' || 'Marketing Manager');

elsif job_id = 'MK_REP' then dbms_output.put_line(e_name || ' ' || 'Marketing
Representative');

elsif job_id = 'HR_REP' then dbms_output.put_line(e_name || ' ' || 'Human Resources
Representative');

else dbms_output.put_line(e_name || ' ' || 'Public Relations Representative');

end if;

end;

/
```



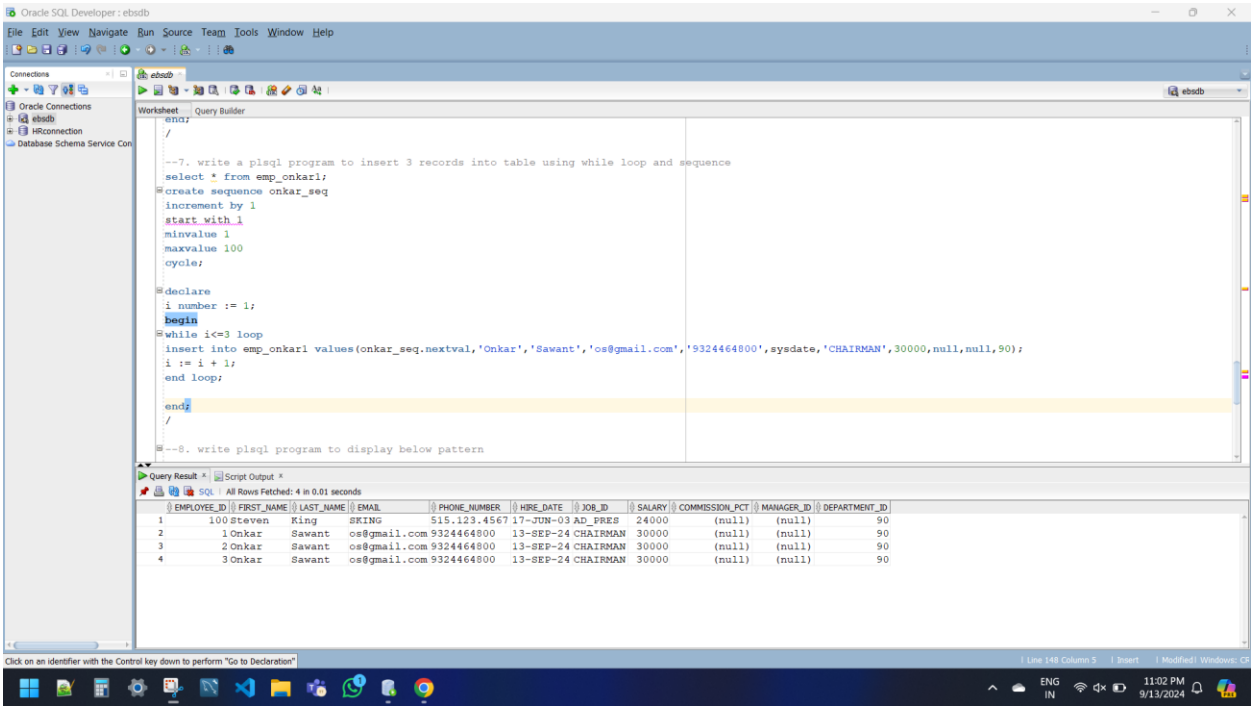

```
--6. write a plsql program to display 25 to 50 using basic loop

declare

i number := 25;

begin

loop
```

```
exit when i >50;
dbms_output.put_line(i);
i := i + 1;
end loop;
end;
/
```




```
--7. write a plsql program to insert 3 records into table using while loop and sequence
select * from emp_onkar1;
create sequence onkar_seq
increment by 1
start with 1
minvalue 1
maxvalue 100
cycle;
```

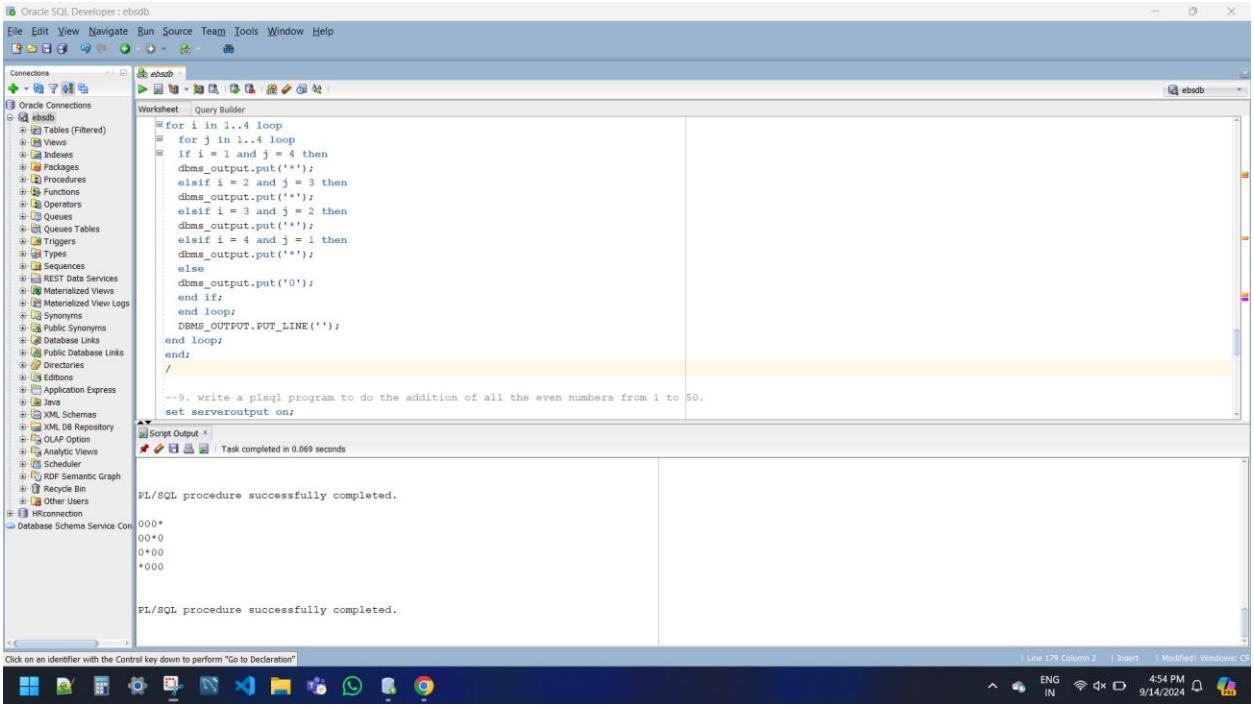
```
declare

i number := 1;

begin

while i<=3 loop

insert into emp_onkar1
values(onkar_seq.nextval,'Onkar','Sawant','os@gmail.com','9324464800',sysdate,'CHAIR
MAN',30000,null,null,90);

i := i + 1;

end loop;


end;

/
```



--8. write plsql program to display below pattern

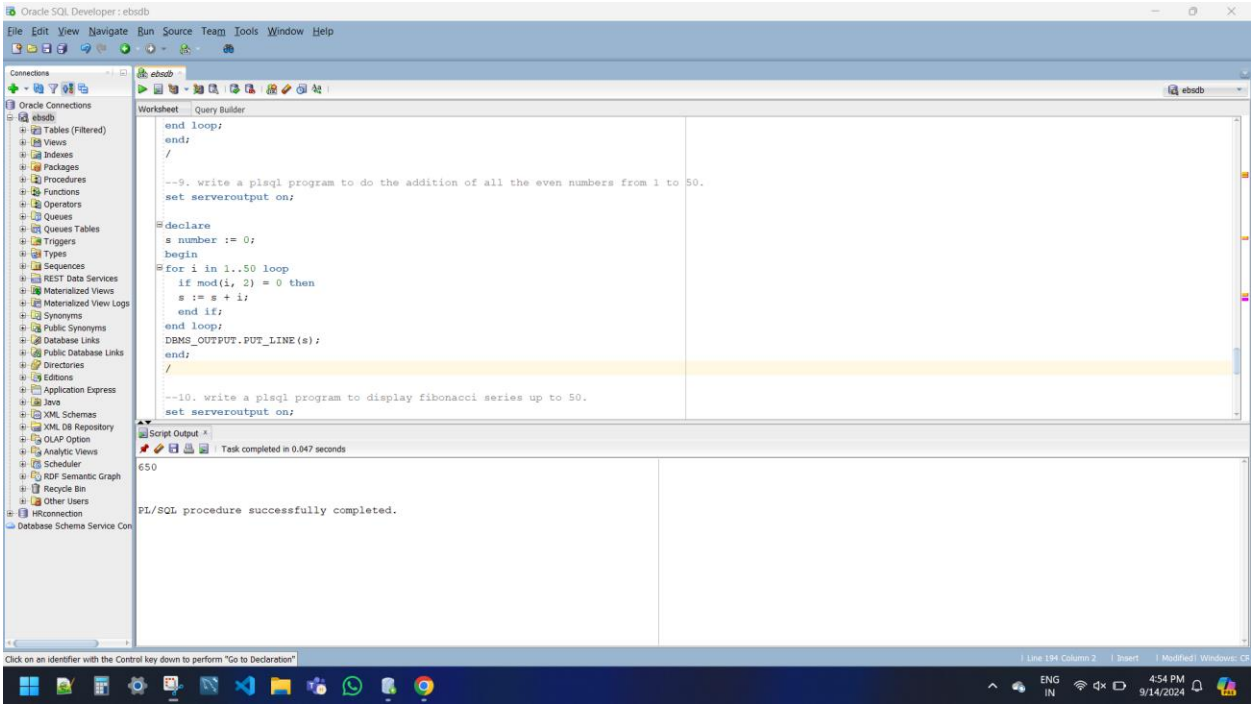
--

```
--    *
--   *
--  *
-- *
set serveroutput on;
begin
for i in 1..4 loop
 for j in 1..4 loop
 if i = 1 and j = 4 then
 dbms_output.put('*');
 elsif i = 2 and j = 3 then
 dbms_output.put('*');
 elsif i = 3 and j = 2 then
 dbms_output.put('*');
 elsif i = 4 and j = 1 then
 dbms_output.put('*');
 else
 dbms_output.put('0');
 end if;
 end loop;
 DBMS_OUTPUT.PUT_LINE('');
end loop;
end;
/
```



--9. write a plsql program to do the addition of all the even numbers from 1 to 50.

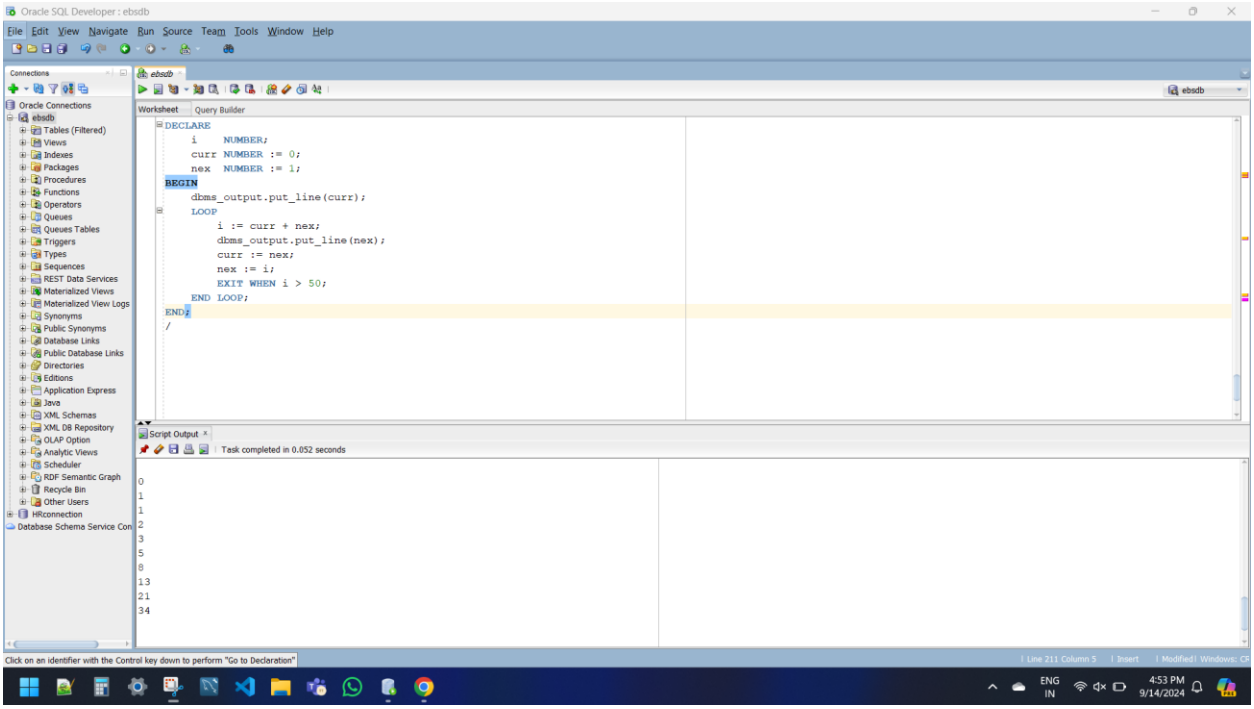
```
set serveroutput on;


declare
s number := 0;
begin
for i in 1..50 loop
  if mod(i, 2) = 0 then
  s := s + i;
  end if;
end loop;
DBMS_OUTPUT.PUT_LINE(s);
end;
/
```



--10. write a plsql program to display fibonacci series up to 50.

```
set serveroutput on;
DECLARE
   i   NUMBER;
   curr NUMBER := 0;
   nex  NUMBER := 1;
BEGIN
   dbms_output.put_line(curr);
   LOOP
      i := curr + nex;
      dbms_output.put_line(nex);
      curr := nex;
      nex := i;
      EXIT WHEN i > 50;
```

```
    END LOOP;

END;

/
```


Oracle SQL Developer : ebsdb
File Edit View Navigate Run Source Team Tools Window Help
Connections
Oracle Connections
ebsdb
Tables (Filtered)
Views
Indexes
Packages
Procedures
Functions
Operators
Queues
Queues Tables
Triggers
Types
Sequences
REST Data Services
Materialized Views
Materialized View Logs
Synonyms
Public Synonyms
Database Links
Public Database Links
Directories
Editions
Application Express
Java
XML Schemas
XML DB Repository
OLAP Option
Analytic Views
Scheduler
RDF Semantic Graph
Recycle Bin
Other Users
HRconnection
Database Schema Service Con
ebsdb
Worksheet
Query Builder
DECLARE
i NUMBER;
curr NUMBER := 0;
nex NUMBER := 1;
BEGIN
dbms_output.put_line(curr);
LOOP
i := curr + nex;
dbms_output.put_line(nex);
curr := nex;
nex := i;
EXIT WHEN i > 50;
END LOOP;
END;
/
Script Output
Task completed in 0.052 seconds
0
1
1
2
3
5
8
13
21
34
Click on an identifier with the Control key down to perform "Go to Declaration"
ENG
IN
4:53 PM
9/14/2024