// Write a Pl/Sql code that shows the department id,location_id, street_address, city where department_id =90.

```
DECLARE
    did loc.DEPARTMENT_ID%type;
    dloc loc.location_id%type;
    add loc.street_address%type;
    city loc.city%type;
BEGIN
    SELECT department_id,location_id,street_address,city INTO
did,dloc,add,city
    FROM loc
    WHERE DEPARTMENT_ID= 90;
   dbms_output.put_line('The details of location  is: ');
   dbms_output.put_line('Department Id  : ' || did);
   dbms_output.put_line('location id :  ' || dloc);
   dbms_output.put_line('street address :  ' || add);
   dbms_output.put_line('city :  ' || city);
END;
```

> Write a code that shows first name, lastname,department where first name = “Neena”.

```
DECLARE
    EFN A.FIRST_NAME%TYPE;
    ELN A.LAST_NAME%TYPE;
    EDP A.DEPARTMENT_NAME%TYPE;
BEGIN
    SELECT FIRST_NAME,LAST_NAME,DEPARTMENT_NAME INTO EFN,ELN,EDP
    FROM A
    WHERE FIRST_NAME='Neena';
```

```
   dbms_output.put_line('The details if Employee is: ');
   dbms_output.put_line('First Name      : ' || EFN);
   dbms_output.put_line('Last Name       :  ' || ELN);
   dbms_output.put_line('Department Name :  ' || EDP);
END;
```

- Create a view

```
CREATE OR REPLACE FORCE VIEW "A" ("FIRST_NAME", "LAST_NAME",
"DEPARTMENT_NAME") AS
  select first_name,last_name,departments.department_name
from employees,departments
where employees.department_id = departments.department_id and
employee_id=101
/
CREATE OR REPLACE FORCE VIEW  "LOC" ("DEPARTMENT_ID", "LOCATION_ID",
"STREET_ADDRESS", "CITY") AS
  select department_id,locations.location_id,street_address,city
from locations,departments
where locations.location_id = departments.location_id and
department_id = 90
/
```

- Loop

```
DECLARE
    i number := 1;
BEGIN
   LOOP
      dbms_output.put_line(i);
      i:=i+1;
      IF i>50 THEN
```

```
         exit;
      END IF;
   END LOOP;
END;
```

- For loop

```
DECLARE
    i number := 1;
    j number := 1;
BEGIN
      <<outer_loop>>
      FOR i IN 1..5 LOOP
         <<inner_loop>>
      FOR j IN 1..5 LOOP
        dbms_output.put_line('i is '|| i || ' j is '||j);
      END LOOP inner_loop;
      END LOOP outer_loop;
END;
```

- While loop

```
DECLARE
    i number := 1;
    j number := 1;
BEGIN
     WHILE i<=50 LOOP
        dbms_output.put_line('i is '|| i);
        i:= i+1;
      END LOOP;
END;
```

- Prime number check

```
DECLARE
 INPUT NUMBER:=:INP;
    FLAG NUMBER := 0;
    I NUMBER :=2;
BEGIN
   IF INPUT<2 THEN
     FLAG := 1;
    END IF;
   FOR I IN 2..INPUT/2 LOOP
     IF I MOD 2 = 0 THEN
         FLAG := 1;
            EXIT;
        END IF;
    END LOOP;
    IF FLAG = 0 THEN
     dbms_output.put_line(INPUT || ' IS A PRIME NUMBER ');
    ELSE
        dbms_output.put_line(INPUT || ' IS NOT A PRIME NUMBER ');
    END IF;
END;
```

- Sum of first 15 prime number (There may be error)

```
DECLARE
 --INPUT NUMBER:=47;
    FLAG NUMBER := 0;
    I NUMBER :=2;
    SUM NUMBER :=0;
    J NUMBER :=1;
```

```
BEGIN
    FOR J IN 1..47 LOOP
        I :=2;
       WHILE I<=J/2 LOOP
      IF I MOD 2 = 0 THEN
          FLAG := 1;
             EXIT;
         END IF;
         I := I+1;
     END LOOP;

     IF FLAG = 0 THEN
      SUM := SUM+J;
      END IF;
        FLAG :=0;
    END LOOP;
    dbms_output.put_line('Sum of first 15 prime numbers are : '|| SUM);
END;
```

- ➢ Procedure – Summation

```
DECLARE
    a number:=:in;
    b number:=:in1;
    c number;

    procedure summation(x IN number, y IN number, z out number) IS
    BEGIN
```

```
        if x>y then
            z:= y;
        else
            z:=y-x;
        end if;
    END;
BEGIN
    summation(a,b,c);
    dbms_output.put_line('The result is : '|| c);
END;
```

- ➢ Cursor

```
DECLARE
     FN EMPLOYEES.FIRST_NAME%TYPE;
     JOB EMPLOYEES.JOB_ID%TYPE;
     SALARY EMPLOYEES.SALARY%TYPE;

     CURSOR EMP IS
         SELECT FIRST_NAME,JOB_ID,SALARY
         FROM EMPLOYEES;
BEGIN
    OPEN EMP;
    LOOP
       FETCH EMP INTO FN,JOB,SALARY;
       EXIT WHEN EMP%notfound;
       dbms_output.put_line('Employee name  : ' || FN);
       dbms_output.put_line('Employee job id  : ' || JOB);
       dbms_output.put_line('Employee salary  : ' || SALARY);
    END LOOP;
    CLOSE EMP;
END;
```

- ➢ Procedure

```
create procedure ddd as
begin
   declare
      a number;
   begin
       a:=12;
       dbms_output.put_line(a);
   end;
end;
```

- ➢ Procedure + Cursor

```
DECLARE
  salcom NUMBER;
  CURSOR cemp IS
  Select * from employees where commission_pct is not null;
  emp employees%rowtype;
  procedure check_commission(salary  IN NUMBER , commssion IN NUMBER)
IS
 BEGIN
 salcom := (emp.salary+ emp.salary*emp.commission_pct);
 dbms_output.put_line (' Salary : ' || salcom);
 END;
BEGIN
 OPEN cemp;
LOOP
 FETCH cemp into emp;
 Exit when cemp%notfound;

 BEGIN

check_commission(emp.salary,emp.commission_pct);

END;

END LOOP;

 CLOSE cemp;

END;
```