1)Declare

```
CURSOR  det_employee IS

SELECT * FROM EMPLOYEE ;
```

BEGIN

```
FOR emp_rec IN det_employee LOOP

     dbms_output.Put_line('ID: ' ||emp_rec.empid ||'name '
||emp_rec.empname ||'job: ' ||emp_rec.job ||'departnment number: '
||emp_rec.deptno ||'manager: ' ||emp_rec.manager ||'salary: '
||emp_rec.sal);

END LOOP;
```

END;

/

```
SQL> Declare
  2  CURSOR  det_employee IS
  3  SELECT * FROM EMPLOYEE ;
  4  BEGIN
  5   FOR emp_rec IN det_employee LOOP
  6    dbms_output.Put_line('ID: ' ||emp_rec.empid ||'name ' ||emp_rec.empname ||'job: ' ||emp_rec.j
ob ||'departnment number: ' ||emp_rec.deptno ||'manager: ' ||emp_rec.manager ||'salary: ' ||emp_rec.
sal);
  7  END LOOP;
  8  END;
  9  /
ID: E0001name Abeyjob: Testerdepartnment number: D004manager: E0004salary: 30000
ID: E0002name Jestojob: Analystdepartnment number: D002manager: E0001salary: 25000
ID: E0003name Adarshjob: Clerkdepartnment number: D004manager: E0004salary: 20000
ID: E0004name Kevinjob: Admindepartnment number: D002manager: E0005salary: 20000
ID: E0005name Bonyjob: Managerdepartnment number: D001manager:  salary: 50000
ID: E0006name Manujob: Supplierdepartnment number: D003manager: E0001salary: 5000

PL/SQL procedure successfully completed.
```

2] create table person(fname varchar(20),lname varchar(20),person_id number);

Insert into person values(‘Noah’,’Centineo’,48);

Insert into person values(‘Zayn’,’Malik’,55);

Insert into person values(‘Selena’,’Gomez’,51);

Insert into person values(‘Gigi’,’Hadid’,47);

```
DECLARE
        CURSOR disp_name IS
        SELECT fname,lname from person WHERE person_id>50;
BEGIN
      FOR i IN disp_name LOOP
            dbms_output.Put_line(i.fname||' '||i.lname);
      END LOOP;
END;
/
```

```
SQL> DECLARE
  2   CURSOR disp_name IS
  3   SELECT fname,lname from person WHERE person_id>50;
  4  BEGIN
  5   FOR i IN disp_name LOOP
  6    dbms_output.Put_line(i.fname||' '||i.lname);
  7  END LOOP;
  8  END;
  9  /
Zayn Malik
Selena Gomez

PL/SQL procedure successfully completed.

Commit complete.
```

3)
```
DECLARE
   my_record  employee %ROWTYPE;
   CURSOR c1 (max_wage NUMBER) IS
      SELECT * FROM employee WHERE sal< max_wage;
 BEGIN
   OPEN c1(200000);
   LOOP
      FETCH c1 INTO my_record;
```

```
    EXIT WHEN c1%NOTFOUND;
    DBMS_OUTPUT.PUT_LINE('Name = ' || my_record.empname || ', salary = '
      || my_record.sal);
  END LOOP;
  CLOSE c1;
END;
/
```

```
SQL> DECLARE
  2      my_record  employee %ROWTYPE;
  3      CURSOR c1 (max_wage NUMBER) IS
  4          SELECT * FROM employee WHERE sal< max_wage;
  5  BEGIN
  6      OPEN c1(200000);
  7      LOOP
  8          FETCH c1 INTO my_record;
  9          EXIT WHEN c1%NOTFOUND;
 10          DBMS_OUTPUT.PUT_LINE('Name = ' || my_record.empname || ', salary = '
 11              || my_record.sal);
 12      END LOOP;
 13      CLOSE c1;
 14  END;
 15  /
Name = Abey, salary = 30000
Name = Jesto, salary = 25000
Name = Adarsh, salary = 20000
Name = Kevin, salary = 20000
Name = Bony, salary = 50000
Name = Manu, salary = 5000

PL/SQL procedure successfully completed.

Commit complete.
```

4)create table accmaster(acc_no number,name varchar(20),balance number);

Create table acctran(acc_no number,trans_date date,deb_cred number,amount number,processed number);

Insert into accmaster values(‘1’,’Centineo’, 15000);

Insert into accmaster values(‘2’,’Zayn’, 32000);

Insert into accmaster values(‘3’,’Sara’, 51000);

Insert into accmaster values(‘4’,’Selina’, 29000);

Insert into acctran values(‘1’,’20-jan-2023’, 0,50000,1);

```
Insert into acctran  values('2','25-jan-2023', 0,50000,0);
Insert into acctran  values('3','28-jan-2023', 1,50000,0);
Insert into acctran  values('4','2-jan-2023', 1,50000,1);
DECLARE
    CURSOR t1 IS
    SELECT acctran.acc_no,deb_cred,amount,processed,balance
    FROM acctran inner join accmaster on acctran.acc_no = accmaster.acc_no
    where processed LIKE '0';
    trecord t1%ROWTYPE;
    newbal number;
 BEGIN
    OPEN t1;
    LOOP
    FETCH t1 INTO trecord;
    EXIT WHEN t1%NOTFOUND;
    update acctran set processed='1';
    if (trecord.deb_cred = '1')
       then
       newbal:=trecord.balance+trecord.amount;
       update accmaster set balance=newbal where trecord.acc_no=accmaster.acc_no ;
       dbms_output.put_line(trecord.balance);
    else
       newbal:=trecord.balance-trecord.amount;
       update accmaster set balance=newbal where trecord.acc_no=accmaster.acc_no ;
       dbms_output.put_line(trecord.balance);
```

```
   end if;

   END LOOP;

   CLOSE t1;

END;

/
```

```
Commit complete.
SQL> DECLARE
  2      CURSOR t1 IS
  3      SELECT acctran.acc_no,deb_cred,amount,processed,balance
  4      FROM acctran inner join accmaster on acctran.acc_no = accmaster.acc_no
  5      where processed LIKE '0';
  6      trecord t1%ROWTYPE;
  7      newbal number;
  8  BEGIN
  9      OPEN t1;
 10      LOOP
 11      FETCH t1 INTO trecord;
 12      EXIT WHEN t1%NOTFOUND;
 13      update acctran set processed='1';
 14      if (trecord.deb_cred = '1')
 15          then
 16          newbal:=trecord.balance+trecord.amount;
 17          update accmaster set balance=newbal where trecord.acc_no=accmaster.acc_no ;
 18          dbms_output.put_line(trecord.balance);
 19      else
 20          newbal:=trecord.balance-trecord.amount;
 21          update accmaster set balance=newbal where trecord.acc_no=accmaster.acc_no ;
 22          dbms_output.put_line(trecord.balance);
 23      end if;
 24      END LOOP;
 25      CLOSE t1;
 26  END;
 27  /
32000
51000

PL/SQL procedure successfully completed.

Commit complete.
SQL>
```

```
Commit complete.
SQL> DECLARE
  2      CURSOR t1 IS
  3      SELECT acctran.acc_no,deb_cred,amount,processed,balance
  4      FROM acctran inner join accmaster on acctran.acc_no = accmaster.acc_no
  5      where processed LIKE '0';
  6      trecord t1%ROWTYPE;
  7      newbal number;
  8  BEGIN
  9      OPEN t1;
 10      LOOP
 11      FETCH t1 INTO trecord;
 12      EXIT WHEN t1%NOTFOUND;
 13      update acctran set processed='1';
 14      if (trecord.deb_cred = '1')
 15          then
 16          newbal:=trecord.balance+trecord.amount;
 17          update accmaster set balance=newbal where trecord.acc_no=accmaster.acc_no ;
 18          dbms_output.put_line(trecord.balance);
 19      else
 20          newbal:=trecord.balance-trecord.amount;
 21          update accmaster set balance=newbal where trecord.acc_no=accmaster.acc_no ;
 22          dbms_output.put_line(trecord.balance);
 23      end if;
 24      END LOOP;
 25      CLOSE t1;
 26  END;
 27  /
32000
51000

PL/SQL procedure successfully completed.

Commit complete.
```