**Ayush Goyal**

**190905522 CSE D 62**

**DBS Lab End Semester Examination**

**Create the following Vehicle Service database which store the different parts details and parts used in the service of different customer vehicles:**

**Schema:**

**Part (PartNo, PartName, VehicleType, UnitPrice, SalesPrice)**

**Service (ServiceNo, PartNo, ServiceDate, CustomerNo, Qty)**

**Constraints: i. Vehicle Type should be ‘V1’, ‘V2’, ‘V3’, ‘V4’, or ‘V5’**

**ii. UnitPrice and SalesPrice should be greater than zero.**

**iii. PartNo, FK in Service refers to PartNo, PK in Part**

**CODE: (DDL)**

The following code is for creating and populating the tables as given in the question.

drop table Part;

drop table Service;

create table Part(

    PartNo number(2),

    PartName varchar(2),

    VehicleType varchar(2),

    UnitPrice number(5),

    SalesPrice number(5),

    primary key(PartNo),

    check(VehicleType in ('V1','V2','V3','V4','V5')),

    check( UnitPrice > 0 and SalesPrice > 0));

create table Service(

    ServiceNo number(2),

    PartNo number(2),

    ServiceDate date,

    CustomerNo varchar(2),

    Qty number(5),

    primary key(ServiceNo,PartNo),

    foreign key (PartNo) references Part(PartNo));

insert into Part values(1, 'P1', 'V1', 10, 12);

insert into Part values(2, 'P2', 'V2', 100, 110);

insert into Part values(3, 'P3', 'V1', 150, 175);

insert into Part values(4, 'P4', 'V3', 200, 250);

insert into Part values(5, 'P5', 'V2', 75, 90);

insert into Service values(1, 1, '01-Jan-17', 'C1', 5);

insert into Service values(1, 3, '01-Jan-17', 'C1', 4);

insert into Service values(2, 3, '05-Feb-18', 'C2', 10);

insert into Service values(3, 1, '15-May-18', 'C3', 9);

insert into Service values(4, 1, '03-Jun-19', 'C1', 5);

After the creation of tables and insertion of data this is the table data:

**A picture containing text

Description automatically generated**

**QUERIES:**

1. **Write the following queries in SQL:**
2. **List the Part Names which are not used to service the vehicle of any customer.**

**CODE:**

(select PartName

 from Part)

minus

(select PartName

 from Service natural join Part);

**OUTPUT:**

**Shape

Description automatically generated with medium confidence**

1. **List the customer number who has got his vehicle serviced maximum number of times.**

**CODE:**

with customerCount(CustomerNo, cnt) as (select CustomerNo, count(\*) as cnt

                                        from Service group by CustomerNo),

     customerMaxCount(val) as (select max(cnt)

                          from customerCount)

select CustomerNo

from customerCount,customerMaxCount

where cnt = val;

**OUTPUT:**

**Text

Description automatically generated**

1. **List the customer number whose vehicle service used all the parts of vehicle type V1.**

**CODE:**

select unique(ser.CustomerNo)

from Service ser

where not exists (

        (select PartNo

         from Part

         where VehicleType = 'V1')

        minus

        (select unique PartNo

         from Service

         where CustomerNo = ser.CustomerNo));

**OUTPUT:**

**A picture containing text

Description automatically generated**

1. **Write a PL/SQL program to find the total profit done in the sales of a given part number in the service of different customer vehicles. Raise an exception for invalid part number.**

**CODE:**

set serveroutput on;

declare

    Invalid\_PartNo Exception;

    cursor c1 is select \* from Part natural join Service;

    cursor c2 is select \* from Part;

    partn Service.PartNo%TYPE;

    profit Part.SalesPrice%TYPE := 0;

    countPart number :=0;

begin

    partn:=&partNo;

    for q in c2

        loop

        if q.PartNo = partn then

        countPart:=countPart+1;

        end if;

        end loop;

    if countPart = 0 then

    raise Invalid\_PartNo;

    end if;

    for p in c1

        loop

        if p.PartNo = partn then

        profit := profit + (p.SalesPrice - p.UnitPrice) \* p.Qty;

        end if;

        end loop;

    dbms\_output.put\_line('Total Profit for part : '||profit);

    Exception

        when Invalid\_PartNo then

            dbms\_output.put\_line('Invalid part number');

        when others then

            dbms\_output.put\_line('Error');

end;

/

**OUTPUT:**

**Text

Description automatically generated**

**Shape

Description automatically generated with medium confidence**

**THE END**