**TASK 1.**

All the tables

**Table 1. – Shopping\_history**

create table shopping\_history(

product varchar not null,

quantity integer not null,

unit\_price integer not null

);

insert into shopping\_history values('milk',3,10),('bread',7,3),('bread',5,2);

**Table 2. – phones**

create table phones(

name varchar(20) not null unique,

phone\_number integer not null unique

);

insert into phones values('Jack',1234),('Lena',3333),('Mark',9999),('Anna',7582);

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Table 3. - calls**

create table calls(

id integer not null,

caller integer not null,

callee integer not null,

duration integer not null,

unique(id)

);

insert into calls values(25,1234,7582,8),(7,9999,7582,1),(18,9999,3333,4),

(2,7582,3333,3),(3,3333,1234,1),(21,3333,1234,1);

**Table 4. - phones1**

create table phones1 like phones;

insert into phones1 values('Jhon',6356),('Addison',4315),('Kate',8003),('Ginny',9831);

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Table 5. – calls1**

create table calls1 like calls;

insert into calls1 values(65,8003,9831,7),(100,9831,8003,3),(145,4315,9831,18);

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Table 6. - transactions**

create table transactions(

amount integer not null,

date date not null);

insert into transactions values(1000,'2020-01-06'),(-10,'2020-01-14'),(-75,'2020-01-20'),(-5,'2020-01-25'),

(-4,'2020-01-29'),(2000,'2020-03-10'),(-75,'2020-03-12'),(-20,'2020-03-15'),

(40,'2020-03-15'),(-50,'2020-03-17'),(200,'2020-10-10'),(-200,'2020-10-10');

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Table 7. - transactions1**

create table transactions1 like transactions;

insert into transactions1 values(1,'2020-06-29'),(35,'2020-02-20'),(-50,'2020-02-03'),(-1,'2020-02-26'),

(-200,'2020-08-01'),(-44,'2020-02-07'),(-5,'2020-02-25'),(1,'2020-06-29'),

(1,'2020-06-29'),(-100,'2020-12-29'),(-100,'2020-12-30'),(-100,'2020-12-31');

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Table 8. – transactions2**

create table transactions2 like transactions;

insert into transactions2 values(6000,'2020-04-03'),(5000,'2020-04-02'),(4000,'2020-04-01'),

(3000,'2020-03-01'),(2000,'2020-02-01'),(1000,'2020-01-01');

**Answer Queries**

**Task 1.**

* **total\_price using group by**

select product, sum(quantity\*unit\_price) as total\_price From shopping\_history group by product order by product desc;

* **total\_price using partition by**

select distinct product, sum(quantity\*unit\_price) over(partition by product) as total\_price from shopping\_history;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Task 2.**

**-- Question 1. answer**

select name from phones where phone\_number in

(select caller from (select caller, duration from calls union all select callee, duration from calls)

group by caller having sum(duration)>=10) order by name;

**-- Question2. answer**

select name from phones1 where phone\_number in(select caller from (select caller, duration from calls1 union all

select callee, duration from calls1) group by caller having sum(duration)>=10) order by name;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Task 3.**

**-- Question 1. Answer ( using Group by)**

select

sum(case when(neg\_cnt >=3 and cre\_amt<=-100)

then (tot\_amt+5)

else (tot\_amt)

end)-60 as balance

from (select sum(amount) as tot\_amt,

count(case when(amount<0) then 1 end) as Neg\_cnt,

sum(case when (amount<0)then amount else 0 end ) as cre\_amt

from transactions

group by extract(month from date));

**-- Question 2. answer**

select

sum(case when(neg\_cnt >=3 and cre\_amt<=-100)

then (tot\_amt+5)

else (tot\_amt)

end)-60 as balance

from (select sum(amount) as tot\_amt,

count(case when(amount<0) then 1 end) as Neg\_cnt,

sum(case when (amount<0)then amount else 0 end ) as cre\_amt

from transactions1

group by extract(month from date));

**-- Question 3. answer**

select

sum(case when(neg\_cnt >=3 and cre\_amt<=-100)

then (tot\_amt+5)

else (tot\_amt)

end)-60 as balance

from (select sum(amount) as tot\_amt,

count(case when(amount<0) then 1 end) as Neg\_cnt,

sum(case when (amount<0)then amount else 0 end ) as cre\_amt

from transactions2

group by extract(month from date));

**-- Task 3. Answer using partition by**

select

sum(case when(neg\_cnt >=3 and cre\_amt<=-100)

then (tot\_amt+5)

else (tot\_amt)

end)-60 as balance

from (select distinct extract(month from date),

sum(amount) over(partition by extract(month from date)) as tot\_amt,

count(case when(amount<0) then 1 end) over(partition by extract(month from date))as Neg\_cnt,

sum(case when (amount<0)then amount else 0 end ) over(partition by extract(month from date)) as cre\_amt

from transactions2);