SQL PROJECT – HIRING ABC COMPANY

**--Task 1**  
  
Create or replace table shopping\_history(  
product varchar(30),  
quantity integer,  
unit\_price integer)  
  
insert into shopping\_history values  
('milk',3,10),  
('bread',7,3),  
('bread',5,2)  
  
--Answer as below  
  
select product,sum(quantity\*unit\_price) as total\_price from shopping\_history group by product  
  
...................................................  
**--Task 2**

**Question given 1**  
  
Create database Task2  
use Task2  
  
create or replace table phones (  
name varchar(30) not null unique,  
phone\_number integer not null unique)  
  
create or replace table calls(  
id integer,  
caller integer not null,  
callee integer not null,  
duration integer,  
unique (id))  
  
insert into phones values ('jack',1234),  
('lena',3333),  
('mark',9999),  
('anna',7582)  
  
select \* from phones  
  
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);  
  
with total\_duration\_details as(  
with caller\_callee\_details as  
(select [p.name](http://p.name/) as customer\_name,c.caller as caller,c.duration as duration from phones p left outer join calls c on p.phone\_number = c.caller  
union all  
select [p.name](http://p.name/) as customer\_name,c.callee as callee,c.duration as duration from phones p left outer join calls c on p.phone\_number = c.callee)  
select customer\_name,sum(duration) as total\_duration from caller\_callee\_details group by customer\_name)  
select \* from total\_duration\_details where total\_duration >= 10  
  
.................................................................................  
**Question given 2:**  
create or replace table phones1 (  
name varchar(30) not null unique,  
phone\_number integer not null unique)  
  
create or replace table calls1(  
id integer,  
caller integer not null,  
callee integer not null,  
duration integer,  
unique (id))  
  
insert into phones1 values  
('John',6356),  
('adison',4315),  
('kate',8003),  
('ginny',9831)  
  
select \* from phones  
  
insert into calls1 values(65,8003,9831,7),  
(100,9831,8003,3),  
(145,4315,9831,18)  
  
with total\_duration\_details as(  
with caller\_callee\_details as  
(select [p.name](http://p.name/) as customer\_name,c.caller as caller,c.duration as duration from phones1 p left outer join calls1 c on p.phone\_number = c.caller  
union all  
select [p.name](http://p.name/) as customer\_name,c.callee as callee,c.duration as duration from phones1 p left outer join calls1 c on p.phone\_number = c.callee)  
select customer\_name,sum(duration) as total\_duration from caller\_callee\_details group by customer\_name)  
select \* from total\_duration\_details where total\_duration >= 10  
  
..................................................................................................

**Task 3**  
**Given table 1**  
  
Create or replace table transactions(  
amount integer not null,  
tran\_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,'2020-01-25'),  
(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');  
  
select \* from transactions;  
  
select sum(debit\_amount) + sum(Credit\_amount) + sum(fee) from  
(select to\_char(tran\_date,'MM'),to\_char(tran\_date,'Mon'),  
   sum(case when amount > 0 then amount else 0 end) Debit\_amount,  
   sum(case when amount < 0 then amount else 0 end) Credit\_amount,  
   sum(case when amount < 0 then 1 else 0 end) N\_transaction,  
   case when  
   abs (sum(case when amount < 0 then amount else 0 end)) > 100  
        and  
        (sum(case when amount < 0 then 1 else 0 end)) >= 3 then 0 else -5 end fee  
         
from  
(select \* from transactions  
 union all  
 select \* from missing\_date)  
group by to\_char(tran\_date,'MM'),to\_char(tran\_date,'Mon')  
Order by 1);  
  
below table created to findout  
Create or replace table missing\_date (  
amount integer not null,  
tran\_date date);  
insert into missing\_date values  
         (0,TO\_DATE('2020-01-01','YYYY-MM-DD')),  
         (0,TO\_DATE('2020-02-02','YYYY-MM-DD')),  
          (0,TO\_DATE('2020-03-03','YYYY-MM-DD')),  
          (0,TO\_DATE('2020-04-04','YYYY-MM-DD')),  
          (0,TO\_DATE('2020-05-05','YYYY-MM-DD')),  
          (0,TO\_DATE('2020-06-06','YYYY-MM-DD')),  
          (0,TO\_DATE('2020-07-07','YYYY-MM-DD')),  
          (0,TO\_DATE('2020-08-08','YYYY-MM-DD')),  
          (0,TO\_DATE('2020-09-09','YYYY-MM-DD')),  
          (0,TO\_DATE('2020-10-10','YYYY-MM-DD')),  
          (0,TO\_DATE('2020-11-11','YYYY-MM-DD')),  
          (0,TO\_DATE('2020-12-12','YYYY-MM-DD'))

---**Answer is 2746**

..............................................................................................  
  
**Given table 2**  
same problem solved with different dataset

Create or replace table transactions2 (  
tran\_date date,  
amount integer)  
  
Insert into transactions2 values  
('2020-06-29',1),  
('2020-02-20',35),  
('2020-02-03',-50),  
('2020-02-26',-1),  
('2020-08-01',-200),  
('2020-02-07',-44),  
('2020-02-25',-5),  
('2020-06-29',1),  
('2020-06-29',1),  
('2020-12-29',-100),  
('2020-12-30',-100),  
('2020-12-31',-100);  
  
select \*,to\_char(tran\_date,'mon') from transactions2;  
  
select sum(debit\_amount) + sum(credit\_amount) + sum(fee) from  
(select to\_char(tran\_date,'mon'),  
sum(case when amount > 0 then amount else 0 end )debit\_amount,  
sum(case when amount < 0 then amount else 0 end) credit\_amount,  
case when  
abs (sum(case when amount < 0 then amount else 0 end)) >= 100  
     and  
     sum(case when amount < 0 then 1 else 0 end) >= 3 then 0 else -5 end as fee  
     from  
     (select \* from transactions2  
      union all  
      select \* from transactions2samp\_data)  
     group by 1);  
  
  
---below table created to add missing months into original table and made union all with above query

Create or replace table transactions2samp\_data (  
tran\_date date,  
amount integer);  
  
Insert into transactions2samp\_data values  
('2020-01-01',0),  
('2020-02-01',0),  
('2020-03-01',0),  
('2020-04-01',0),  
('2020-05-01',0),  
('2020-06-01',0),  
('2020-07-01',0),  
('2020-08-01',0),  
('2020-09-01',0),  
('2020-10-01',0),  
('2020-11-01',0),  
('2020-12-01',0)

**Answer is -612**   
.............................................................................  
**Given table 3**  
  
Create or replace table transactions3 (  
tran\_date date,  
amount integer);  
  
Insert into transactions3 values  
('2020-04-03',6000),  
('2020-04-02',5000),  
('2020-04-01',4000),  
('2020-03-01',3000),  
('2020-02-01',2000),  
('2020-01-01',1000);  
  
select sum(debit\_amount)+sum(credit\_amount)+sum(fee) from  
(select to\_char(tran\_date,'MM'),  
sum(case when amount > 0 then amount else 0 end) debit\_amount,  
sum(case when amount < 0 then amount else 0 end) credit\_amount,  
case when  
abs (sum(case when amount < 0 then amount else 0 end)) >= 100  
and  
(sum(case when amount < 0 then 1 else 0 end)) >= 3 then 0 else -5 end fee  
from  
 (select \* from transactions3  
  union all  
  select \* from transactions3samp\_data)  
group by 1);

---below table created to add missing months into original table and made union all with above query

Create or replace table transactions3samp\_data (  
tran\_date date,  
amount integer);  
  
Insert into transactions3samp\_data values  
('2020-01-01',0),  
('2020-02-01',0),  
('2020-03-01',0),  
('2020-04-01',0),  
('2020-05-01',0),  
('2020-06-01',0),  
('2020-07-01',0),  
('2020-08-01',0),  
('2020-09-01',0),  
('2020-10-01',0),  
('2020-11-01',0),  
('2020-12-01',0)

---**Answer is 20,940**