-- TASK 1

CREATE DATABASE INEURON\_ASSIGNMENT;

USE INEURON\_ASSIGNMENT;

CREATE OR REPLACE TABLE SHOPPING\_HISTORY (

PRODUCT VARCHAR(50) NOT NULL,

QUANTITY INT NOT NULL,

UNIT\_PRICE INT NOT NULL);

INSERT INTO SHOPPING\_HISTORY VALUES ('Chips', 3, 20);

INSERT INTO SHOPPING\_HISTORY VALUES ('Bread', 2, 35);

INSERT INTO SHOPPING\_HISTORY VALUES ('Milk', 1, 45);

INSERT INTO SHOPPING\_HISTORY VALUES ('Fruits', 8, 90);

INSERT INTO SHOPPING\_HISTORY VALUES ('Potato', 3, 20);

INSERT INTO SHOPPING\_HISTORY VALUES ('Namkeen', 4, 30);

INSERT INTO SHOPPING\_HISTORY VALUES ('Ketchup', 1, 120);

INSERT INTO SHOPPING\_HISTORY VALUES ('Cauliflower', 3, 25);

INSERT INTO SHOPPING\_HISTORY VALUES ('Chips', 6, 20);

INSERT INTO SHOPPING\_HISTORY VALUES ('Paneer', 1, 320);

select \* from SHOPPING\_HISTORY;

SELECT PRODUCT, SUM(QUANTITY\*UNIT\_PRICE) AS TOTAL FROM SHOPPING\_HISTORY GROUP BY 1 ORDER BY PRODUCT DESC;

-- TASK 2

CREATE TABLE PHONES(

NAME VARCHAR(20) NOT NULL UNIQUE,

phone\_number integer not null unique);

CREATE TABLE CALLS(

id integer not null,

caller integer not null,

callee integer not null,

duration integer not null,

unique(id)

);

INSERT INTO PHONES VALUES

('Jack', 1234),

('Lenna', 3333),

('Mark', 9999),

('Anna', 7582);

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);

SELECT \* FROM PHONES;

SELECT \* FROM CALLS;

WITH FINAL\_TAB AS(

SELECT CALLER, SUM(CALL\_TIME) AS CALLING\_TIME FROM (

SELECT CALLER, SUM(DURATION) AS CALL\_TIME FROM CALLS GROUP BY 1

UNION ALL

SELECT CALLEE, SUM(DURATION) AS CALL\_TIME FROM CALLS GROUP BY 1)

GROUP BY 1 HAVING CALLING\_TIME >= 10)

SELECT P.NAME, F.CALLING\_TIME

FROM FINAL\_TAB F

LEFT OUTER JOIN PHONES P

ON (F.CALLER = P.PHONE\_NUMBER) ORDER BY F.CALLING\_TIME DESC;

-- TASK 3

CREATE TABLE IF NOT EXISTS TRANSACTIONS(

AMOUNT INTEGER NOT NULL,

Date date NOT NULL

);

INSERT INTO transactions (Amount, Date) VALUES (1000, '2020-01-06');

INSERT INTO transactions (Amount, Date) VALUES (-10, '2020-01-14');

INSERT INTO transactions (Amount, Date) VALUES (-75, '2020-01-20');

INSERT INTO transactions (Amount, Date) VALUES (-5, '2020-01-25');

INSERT INTO transactions (Amount, Date) VALUES (-4, '2020-01-29');

INSERT INTO transactions (Amount, Date) VALUES (2000, '2020-03-10');

INSERT INTO transactions (Amount, Date) VALUES (-75, '2020-03-12');

INSERT INTO transactions (Amount, Date) VALUES (-20, '2020-03-15');

INSERT INTO transactions (Amount, Date) VALUES (40, '2020-03-15');

INSERT INTO transactions (Amount, Date) VALUES (-50, '2020-03-17');

INSERT INTO transactions (Amount, Date) VALUES (200, '2020-10-10');

INSERT INTO transactions (Amount, Date) VALUES (-200, '2020-10-10');

SELECT \* FROM TRANSACTIONS;

WITH FINNN\_TAB AS(

WITH FIN\_TAB AS (

select sum(AMOUNT) AS SUM\_AMOUNT, EXTRACT(MONTH FROM DATE) AS MONTH FROM TRANSACTIONS WHERE MONTH IN (

SELECT EXTRACT(MONTH FROM DATE) AS MONTH FROM TRANSACTIONS WHERE AMOUNT < 0

EXCEPT

SELECT MONTH

FROM (select count(amount) AS NUM\_OF\_TXN, EXTRACT(MONTH FROM DATE) AS MONTH from transactions group by 2 having count(amount) > 3) AS TXN

INNER JOIN

(SELECT SUM(AMOUNT) AS SUM\_AMOUNT, EXTRACT(MONTH FROM DATE) AS MONTH FROM TRANSACTIONS WHERE AMOUNT < 0 GROUP BY 2 HAVING SUM\_AMOUNT < -100)

USING (MONTH)) group by 2)

SELECT SUM(SUM\_AMOUNT) AS FIN\_AMOUNT, count(month) AS Mon\_Num FROM FIN\_TAB)

SELECT SUM(FIN\_AMOUNT) - 5\*SUM(Mon\_Num) AS FINAL\_BALANCE FROM FINNN\_TAB;