***SNOWFLAKE\_PROJECT\_ASSIGNMENT-1***

USE DATABASE INEURON\_ASSIGNMENT1;

**/\*\*\*\*TASK1\*\*\*\*/**

CREATE OR REPLACE TABLE "INEURON\_ASSIGNMENT1"."PUBLIC"."SHOPPING\_HISTORY" (

PRODUCT VARCHAR (10) NOT NULL,

QUANTITY INTEGER NOT NULL,

UNIT\_PRICE INTEGER NOT NULL);

SELECT \* FROM SHOPPING\_HISTORY;

INSERT INTO SHOPPING\_HISTORY(PRODUCT,QUANTITY,UNIT\_PRICE) VALUES('MILK',3,10);

INSERT INTO SHOPPING\_HISTORY(PRODUCT,QUANTITY,UNIT\_PRICE) VALUES('BREAD',7,3);

INSERT INTO SHOPPING\_HISTORY(PRODUCT,QUANTITY,UNIT\_PRICE) VALUES('BREAD',5,2);

**SELECT PRODUCT,SUM(QUANTITY\*UNIT\_PRICE) AS TOTAL\_PRICE FROM SHOPPING\_HISTORY**

**GROUP BY 1**

**ORDER BY 1 DESC;**

**/\*\*\*\*TASK2\*\*\*\*/**

**/\*1\*/**

CREATE OR REPLACE TABLE PHONES(

NAME VARCHAR(20) NOT NULL UNIQUE,

PHONE\_NUMBER INTEGER NOT NULL UNIQUE);

CREATE OR REPLACE TABLE CALLS(

ID INTEGER NOT NULL,

CALLER INTEGER NOT NULL,

CALLEE INTEGER NOT NULL,

DURATION INTEGER NOT NULL,

UNIQUE(ID));

INSERT INTO PHONES(NAME,PHONE\_NUMBER) VALUES('Jack',1234);

INSERT INTO PHONES(NAME,PHONE\_NUMBER) VALUES('Lena',3333);

INSERT INTO PHONES(NAME,PHONE\_NUMBER) VALUES('Mark',9999);

INSERT INTO PHONES(NAME,PHONE\_NUMBER) VALUES('Anna',7582);

INSERT INTO CALLS(ID,CALLER,CALLEE,DURATION) VALUES(25,1234,7582,8);

INSERT INTO CALLS(ID,CALLER,CALLEE,DURATION) VALUES(7,9999,7582,1);

INSERT INTO CALLS(ID,CALLER,CALLEE,DURATION) VALUES(18,9999,3333,4);

INSERT INTO CALLS(ID,CALLER,CALLEE,DURATION) VALUES(2,7582,3333,3);

INSERT INTO CALLS(ID,CALLER,CALLEE,DURATION) VALUES(3,3333,1234,1);

INSERT INTO CALLS(ID,CALLER,CALLEE,DURATION) VALUES(21,3333,1234,1);

SELECT \* FROM CALLS;

SELECT \* FROM PHONES;

SELECT P.NAME

FROM PHONES P

INNER JOIN

(SELECT CALLER ,TOTAL\_DURATIONS FROM (SELECT CALLER,SUM(DURATION) AS TOTAL\_DURATIONS FROM (SELECT CALLER, DURATION FROM CALLS

UNION ALL

SELECT CALLEE,DURATION FROM CALLS)

GROUP BY 1

HAVING TOTAL\_DURATIONS>=10

ORDER BY 2 DESC))

ON P.PHONE\_NUMBER=CALLER

ORDER BY P.NAME;

**/\*2\*/**

CREATE OR REPLACE TABLE PHONES(

NAME VARCHAR(20) NOT NULL UNIQUE,

PHONE\_NUMBER INTEGER NOT NULL UNIQUE);

CREATE OR REPLACE TABLE CALLS(

ID INTEGER NOT NULL,

CALLER INTEGER NOT NULL,

CALLEE INTEGER NOT NULL,

DURATION INTEGER NOT NULL,

UNIQUE(ID));

INSERT INTO PHONES(NAME,PHONE\_NUMBER) VALUES('John',6356);

INSERT INTO PHONES(NAME,PHONE\_NUMBER) VALUES('Addison',4315);

INSERT INTO PHONES(NAME,PHONE\_NUMBER) VALUES('Kate',8003);

INSERT INTO PHONES(NAME,PHONE\_NUMBER) VALUES('Ginny',9831);

INSERT INTO CALLS(ID,CALLER,CALLEE,DURATION) VALUES(65,8003,9831,7);

INSERT INTO CALLS(ID,CALLER,CALLEE,DURATION) VALUES(100,9831,8003,3);

INSERT INTO CALLS(ID,CALLER,CALLEE,DURATION) VALUES(145,4315,9831,18);

SELECT P.NAME

FROM PHONES P

INNER JOIN

(SELECT CALLER ,TOTAL\_DURATIONS FROM (SELECT CALLER,SUM(DURATION) AS TOTAL\_DURATIONS FROM (SELECT CALLER, DURATION FROM CALLS

UNION ALL

SELECT CALLEE,DURATION FROM CALLS)

GROUP BY 1

HAVING TOTAL\_DURATIONS>=10))

ON P.PHONE\_NUMBER=CALLER

ORDER BY P.NAME;

**/\*\*\*\*TASK3\*\*\*\*/**

**/\*1\*/**

CREATE OR REPLACE TABLE TRANSACTIONS(

AMOUNT INTEGER NOT NULL,

DATE DATE NOT NULL);

INSERT INTO TRANSACTIONS VALUES(1000,'2020-01-06');

INSERT INTO TRANSACTIONS VALUES(-10,'2020-01-14');

INSERT INTO TRANSACTIONS VALUES(-75,'2020-01-20');

INSERT INTO TRANSACTIONS VALUES(-5,'2020-01-25');

INSERT INTO TRANSACTIONS VALUES(-4,'2020-01-29');

INSERT INTO TRANSACTIONS VALUES(2000,'2020-03-10');

INSERT INTO TRANSACTIONS VALUES(-75,'2020-03-12');

INSERT INTO TRANSACTIONS VALUES(-20,'2020-03-15');

INSERT INTO TRANSACTIONS VALUES(40,'2020-03-15');

INSERT INTO TRANSACTIONS VALUES(-50,'2020-03-17');

INSERT INTO TRANSACTIONS VALUES(200,'2020-10-10');

INSERT INTO TRANSACTIONS VALUES(-200,'2020-10-10');

SELECT \* FROM TRANSACTIONS;

SELECT AMOUNT,EXTRACT('MONTH',DATE) FROM TRANSACTIONS;

SELECT SUM(

CASE

WHEN (EXPENSE<=-100 AND COUNT>=2) THEN AMOUNT

ELSE AMOUNT-5

END) AS BALANCE

FROM

(SELECT SUM(AMOUNT) AS AMOUNT ,DATE, SUM(EXPENSE) AS EXPENSE,SUM(COUNT) AS COUNT FROM

(SELECT AMOUNT ,DATE,

CASE

WHEN AMOUNT<0 THEN AMOUNT

ELSE 0

END AS EXPENSE,

CASE

WHEN AMOUNT<0 THEN 1

ELSE 0

END AS COUNT

FROM

(select AMOUNT,extract('MONTH',DATE) AS DATE from TRANSACTIONS

UNION ALL

select UNIFORM(0,12,0) ,seq4()+1 as months from table (generator(rowcount=>12)) v order by 1))

GROUP BY 2);

**/\*2\*/**

CREATE OR REPLACE TABLE TRANSACTIONS1(

AMOUNT INTEGER NOT NULL,

DATE DATE NOT NULL);

INSERT INTO TRANSACTIONS1 VALUES(1,'2020-06-29');

INSERT INTO TRANSACTIONS1 VALUES(35,'2020-02-20');

INSERT INTO TRANSACTIONS1 VALUES(-50,'2020-02-03');

INSERT INTO TRANSACTIONS1 VALUES(-1,'2020-02-26');

INSERT INTO TRANSACTIONS1 VALUES(-200,'2020-08-01');

INSERT INTO TRANSACTIONS1 VALUES(-44,'2020-02-07');

INSERT INTO TRANSACTIONS1 VALUES(-5,'2020-02-25');

INSERT INTO TRANSACTIONS1 VALUES(1,'2020-06-29');

INSERT INTO TRANSACTIONS1 VALUES(1,'2020-06-29');

INSERT INTO TRANSACTIONS1 VALUES(-100,'2020-12-29');

INSERT INTO TRANSACTIONS1 VALUES(-100,'2020-12-30');

INSERT INTO TRANSACTIONS1 VALUES(-100,'2020-12-31');

SELECT \* FROM TRANSACTIONS1;

SELECT AMOUNT,EXTRACT('MONTH',DATE) FROM TRANSACTIONS1;

SELECT SUM(

CASE

WHEN (EXPENSE<=-100 AND COUNT>=2) THEN AMOUNT

ELSE AMOUNT-5

END) AS BALANCE

FROM

(SELECT SUM(AMOUNT) AS AMOUNT ,DATE, SUM(EXPENSE) AS EXPENSE,SUM(COUNT) AS COUNT FROM

(SELECT AMOUNT ,DATE,

CASE

WHEN AMOUNT<0 THEN AMOUNT

ELSE 0

END AS EXPENSE,

CASE

WHEN AMOUNT<0 THEN 1

ELSE 0

END AS COUNT

FROM

(select AMOUNT,extract('MONTH',DATE) AS DATE from TRANSACTIONS1

UNION ALL

select UNIFORM(0,12,0) ,seq4()+1 as months from table (generator(rowcount=>12)) v order by 1))

GROUP BY 2);

**/\*3\*/**

CREATE OR REPLACE TABLE TRANSACTIONS2(

AMOUNT INTEGER NOT NULL,

DATE DATE NOT NULL);

INSERT INTO TRANSACTIONS2 VALUES(6000,'2020-04-03');

INSERT INTO TRANSACTIONS2 VALUES(5000,'2020-04-02');

INSERT INTO TRANSACTIONS2 VALUES(4000,'2020-04-01');

INSERT INTO TRANSACTIONS2 VALUES(3000,'2020-03-01');

INSERT INTO TRANSACTIONS2 VALUES(2000,'2020-02-01');

INSERT INTO TRANSACTIONS2 VALUES(1000,'2020-01-01');

SELECT \* FROM TRANSACTIONS2;

SELECT AMOUNT,EXTRACT('MONTH',DATE) FROM TRANSACTIONS2;

SELECT SUM(

CASE

WHEN (EXPENSE<=-100 AND COUNT>=2) THEN AMOUNT

ELSE AMOUNT-5

END) AS BALANCE

FROM

(SELECT SUM(AMOUNT) AS AMOUNT ,DATE, SUM(EXPENSE) AS EXPENSE,SUM(COUNT) AS COUNT FROM

(SELECT AMOUNT ,DATE,

CASE

WHEN AMOUNT<0 THEN AMOUNT

ELSE 0

END AS EXPENSE,

CASE

WHEN AMOUNT<0 THEN 1

ELSE 0

END AS COUNT

FROM

(select AMOUNT,extract('MONTH',DATE) AS DATE from TRANSACTIONS2

UNION ALL

select UNIFORM(0,12,0) ,seq4()+1 as months from table (generator(rowcount=>12)) v order by 1))

GROUP BY 2);