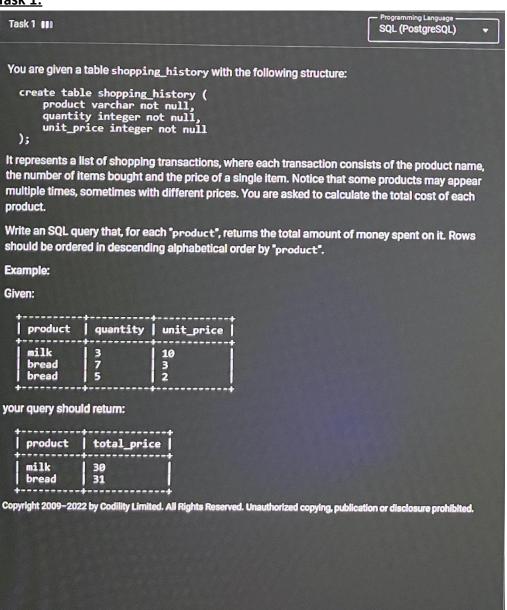
Read the Task Very minutely for all test cases to pass: STRICTLY NO COPY PASTE FROM GOOGLE

Create the following table structure in SNOWFLAKE by creating your own warehouse. Insert some 10 rows using INSERT command (check task 3 and same way insert for all task tables) in the table by trying different values for all the columns and then check using SELECT *

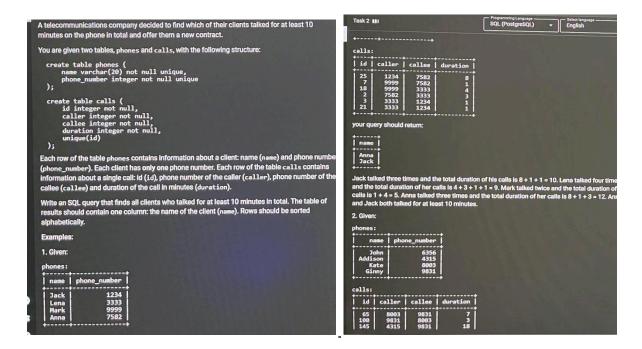
Once data is loaded, performed the below task

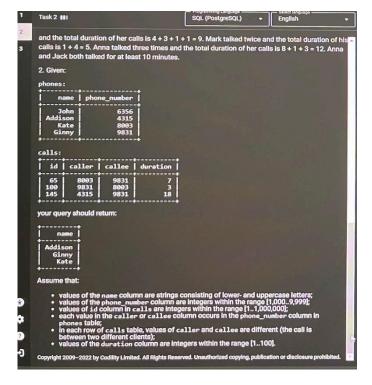
Task 1:



Read the Task Very minutely for all test cases to pass: STRICTLY NO COPY PASTE FROM GOOGLE

Task 2:





Read the Task Very minutely for all test cases to pass: STRICTLY NO COPY PASTE FROM GOOGLE

Task 3: Output display is just one column balance

You are given a history of your bank account transactions for the year 2020. Each transaction was either a credit card payment or an incoming transfer.

There is a fee for holding a credit card which you have to pay every month. The cost you are charged each month is 5. However, you are not charged for a given month if you made at least three credit card payments for a total cost of at least 100 within that month. Note that this fee is not included in the supplied history of transactions.

At the beginning of the year, the balance of your account was 0. Your task is to compute the balance at the end of the year.

You are given a table transactions with the following structure:

```
create table transactions (
    amount integer not null,
    date date not null
);
```

Each row of the table contains information about a single transaction: the amount of money (amount) and the date when the transaction happened (date). If the amount value is negative, it is a credit card payment. Otherwise, it is an incoming transfer. There are no transactions with an amount of 0.

Write an SQL query that returns a table containing one column, balance. The table should contain one row with the total balance of your account at the end of the year, including the fee for holding a credit card.

Examples:

1. Given table:

date	amount
2020-01-06	1000
2020-01-14	-10
2020-01-20	-75
2020-01-25	-5
2020-01-29	-4
2020-03-10	2000
2020-03-12	-75
2020-03-15	-20
2020-03-15	40
2020-03-17	-50
2020-10-10	200
2020-10-10	-200

your query should return:

+	balance	+
1	2746	+

The balance without the credit card fee would be 2801. You are charged a fee for every month except March, which in total equates to 11 * 5 = 55.

your query should return:

+	
Ï	balance
+	
1	2746
+	

The balance without the credit card fee would be 2801. You are charged a fee for every month except March, which in total equates to 11 * 5 = 55.

In March, you had three transactions for a total cost of 75 + 20 + 50 = 145, thus you are not charged the fee. In January, you had four card payments for a total cost of 10 + 75 + 5 + 4 = 94, which is less than 100; thus you are charged. In October, you had one card payment for a total cost of 200 but you need to have at least three payments in a month; thus you are charged. In all other months (February, April, ...) you had no card payments, thus you are charged.

The final balance is 2801 - 55 = 2746.

2. Given table:

amount	date
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

your query should return:

+	
1	balance
+	
1	-612

The balance excluding the fee would be -562. You are not charged the fee in February since you had four card payments for a total cost of 50 + 1+44+5=100 in that month. You are also not charged the fee in December since you had three card payments for a total cost of 100+100+100=300. The final balance is -562 - 10*5=-612.

3. Given table:

amount	date
6000	1 2020-04-03
5000	2020-04-02
4000	2020-04-01
3000	2020-03-01
2000	2020-02-01
1000	2020-01-01

Read the Task Very minutely for all test cases to pass: STRICTLY NO COPY PASTE FROM GOOGLE

amount	date	!
6000 5000 4000 3000 2000 1000	2020-04-03 2020-04-02 2020-04-01 2020-03-01 2020-02-01 2020-01-01	*
our query s	should return:	
balance	1	
	1	
balance 20940 ou earned	1	e charged a fee for every month. The final 1940.
20940 ou earned alance is 2	21000 but you ar 1000 - 12 * 5 = 2	
20940 ou earned alance is 2 ssume tha	21000 but you ar 1000 - 12 * 5 = 2	

You can add the following data in the table

```
1 V CREATE TABLE transactions(
       Amount INTEGER NOT NULL
 2
 3
      ,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');
10
11 INSERT INTO transactions (Amount, Date) VALUES (-75, '2020-03-12');
12 INSERT INTO transactions(Amount, Date) VALUES (-20, '2020-03-15');
   INSERT INTO transactions (Amount, Date) VALUES (40, '2020-03-15');
13
14 INSERT INTO transactions (Amount, Date) VALUES (-50, '2020-03-17');
    INSERT INTO transactions (Amount, Date) VALUES (200, '2020-10-10');
16 INSERT INTO transactions(Amount, Date) VALUES (-200, '2020-10-10');
17
```