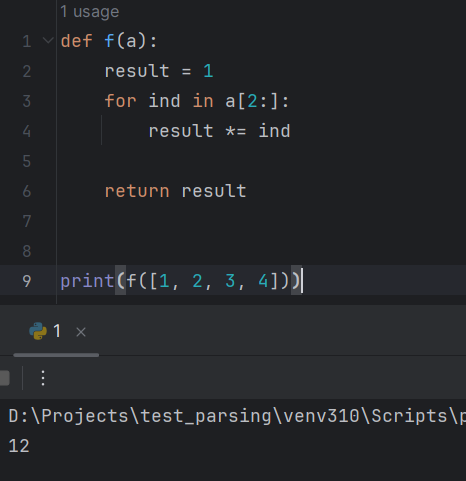
Синякин Даниил Сергеевич

03.07.2024

**ОБЩИЕ ЗАДАЧИ**

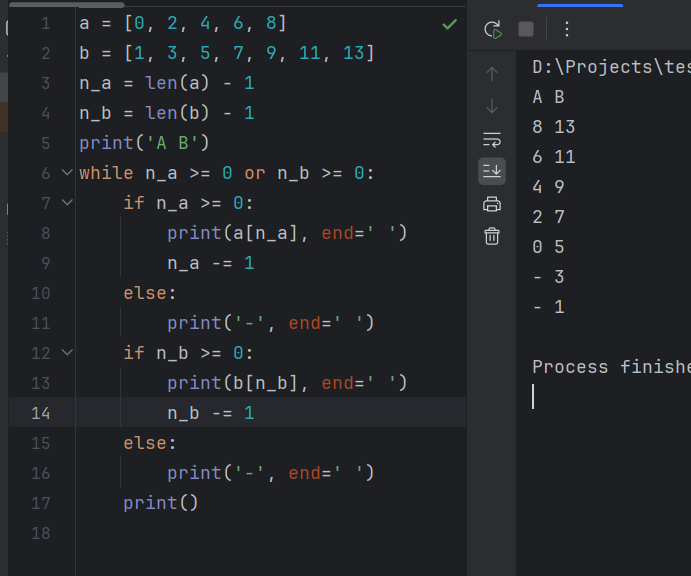
1. def f(a):  
    result = 1  
    for ind in a[2:]:  
    result \*= ind  
     
    return result



перебираем массив со 2 элемента и перемножаем

a = [0, 2, 4, 6, 8]  
b = [1, 3, 5, 7, 9, 11, 13]  
n\_a = len(a) - 1  
n\_b = len(b) - 1  
print('A B')  
while n\_a >= 0 or n\_b >= 0:  
 if n\_a >= 0:  
 print(a[n\_a], end=' ')  
 n\_a -= 1  
 else:  
 print('-', end=' ')  
 if n\_b >= 0:  
 print(b[n\_b], end=' ')  
 n\_b -= 1  
 else:  
 print('-', end=' ')  
 print()

тк могут быть массивы разной длины, то используем цикл while, если равная длина гарантирована, то можно через zip



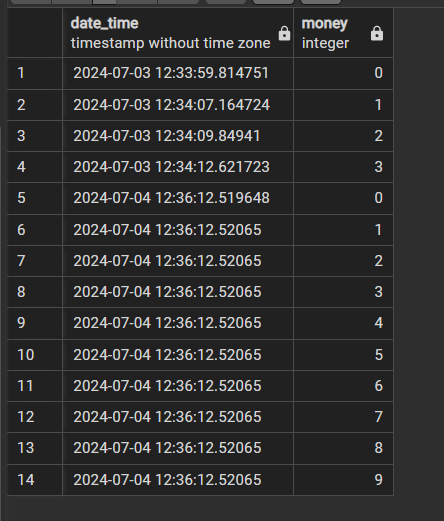
1. Не знаю

SELECT to\_char(date\_time, 'YYYY-MM-DD') as date\_group, sum(money)

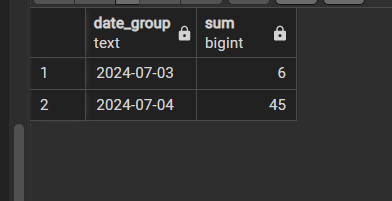
FROM TBL

group by date\_group

Сделал вот такую табличку



Суммируем по дате без времени, результат



import random

n = 11

a = [i for i in range(n + 1)]

random.shuffle(a)

del a[random.randint(0, n - 1)]

print(a)

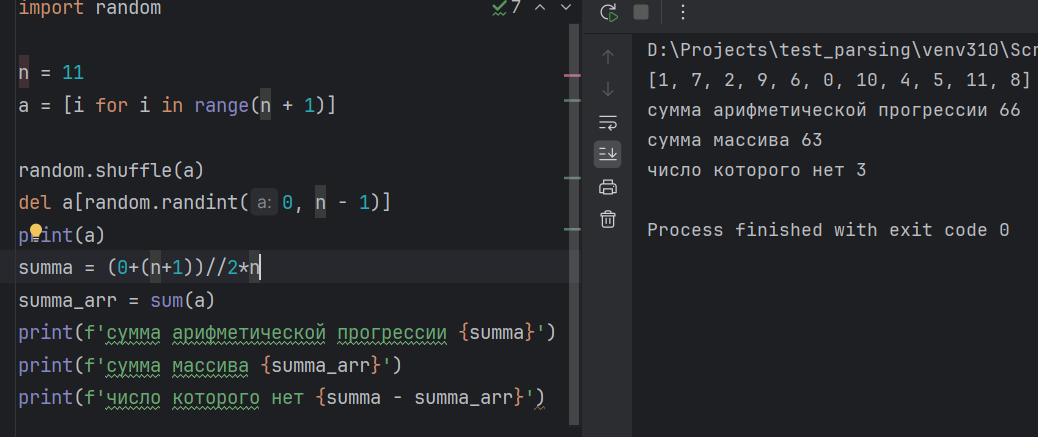
summa = (0+(n+1))//2\*n

summa\_arr = sum(a)

print(f'сумма арифметической прогрессии {summa}')

print(f'сумма массива {summa\_arr}')

print(f'число которого нет {summa - summa\_arr}')

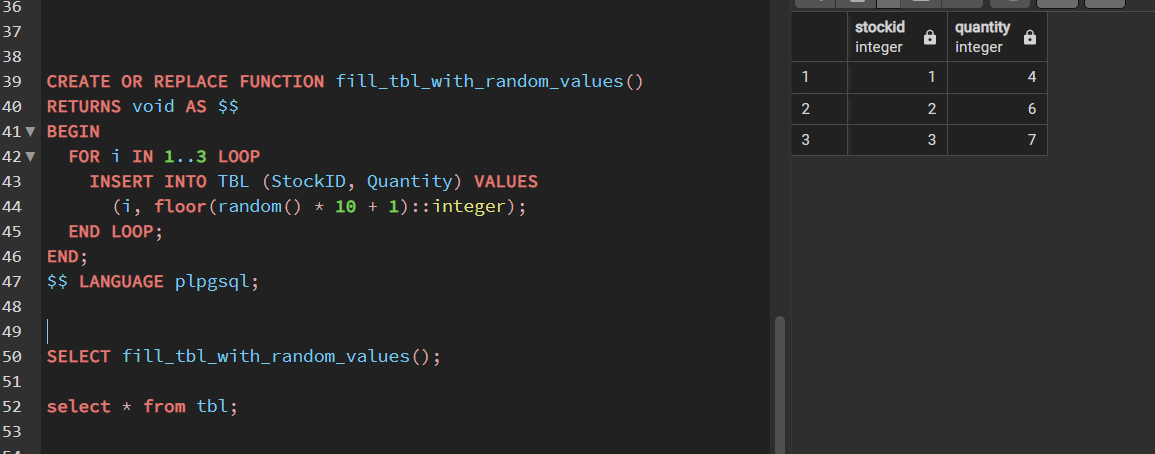


Тк у нас массив задан по формуле арифметической прогрессии, то можно обойтись без цикла

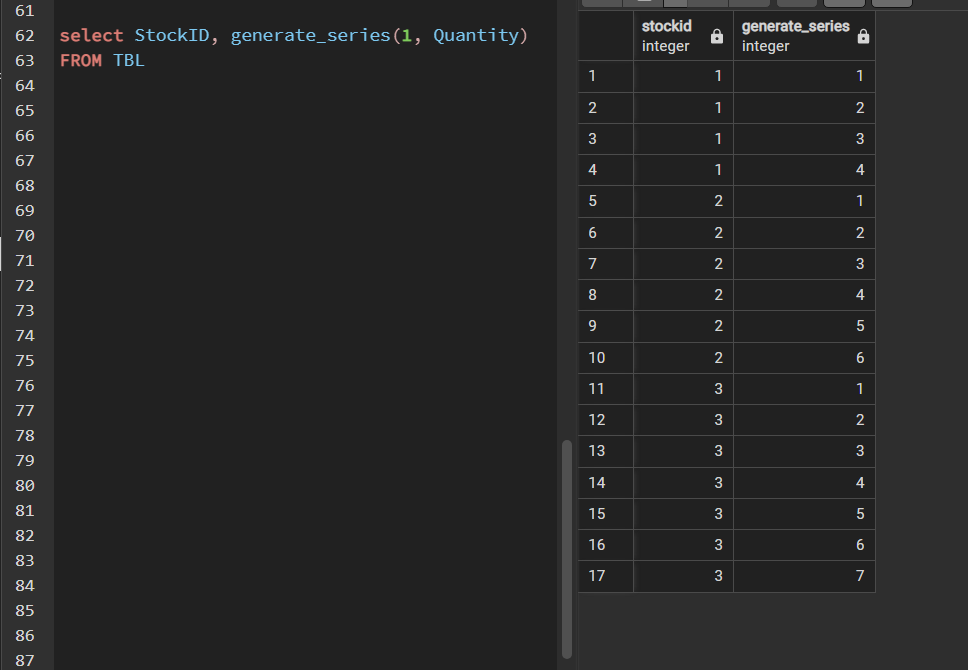
select StockID, generate\_series(1, Quantity)

FROM TBL

С помощью кода заполнил табличку



Результат



factorial\_data = {

0: 1,

1: 1

}

def factorial(n):

if n in factorial\_data:

return factorial\_data[n]

result = n \* factorial(n - 1)

factorial\_data[n] = result

return result

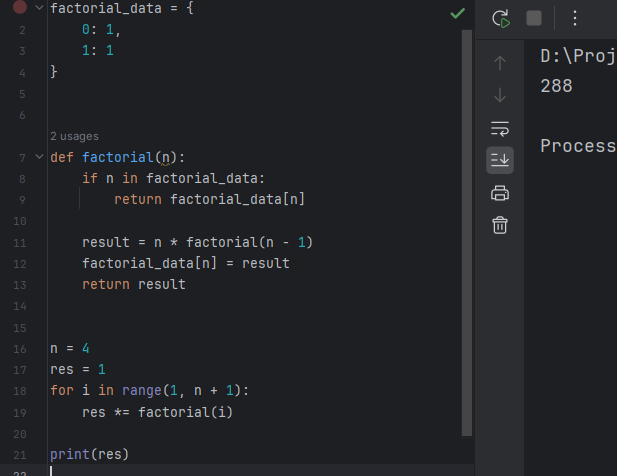
n = 4

res = 1

for i in range(1, n + 1):

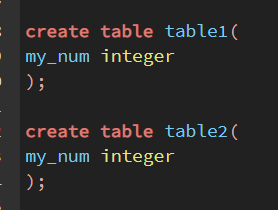
res \*= factorial(i)

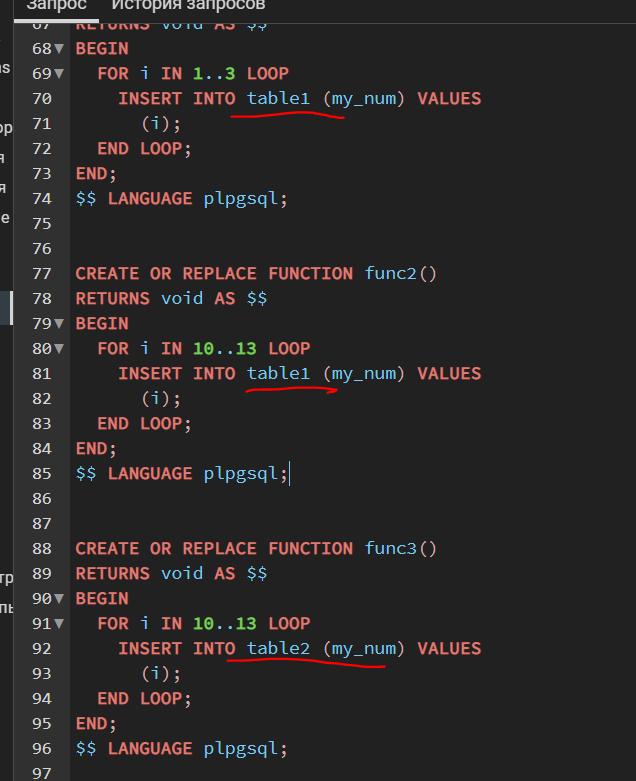
print(res)



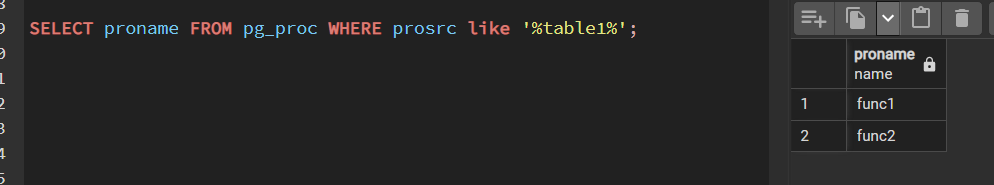
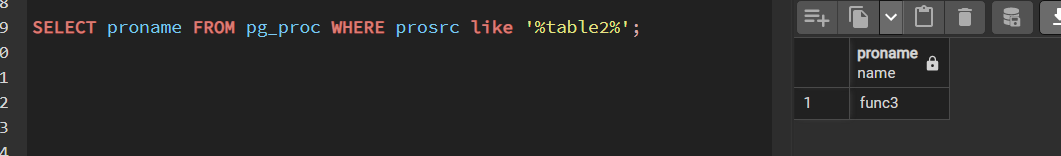
SELECT proname FROM pg\_proc WHERE prosrc like '%table1%';

Сделал 2 таблицы и 3 функции





Результат

WITH summary\_count AS (

SELECT

sum((length(string) - length(replace(string, 'Hello world!', ''))) / length('Hello world!')) as summary

FROM

tbl

)

select

id,

string,

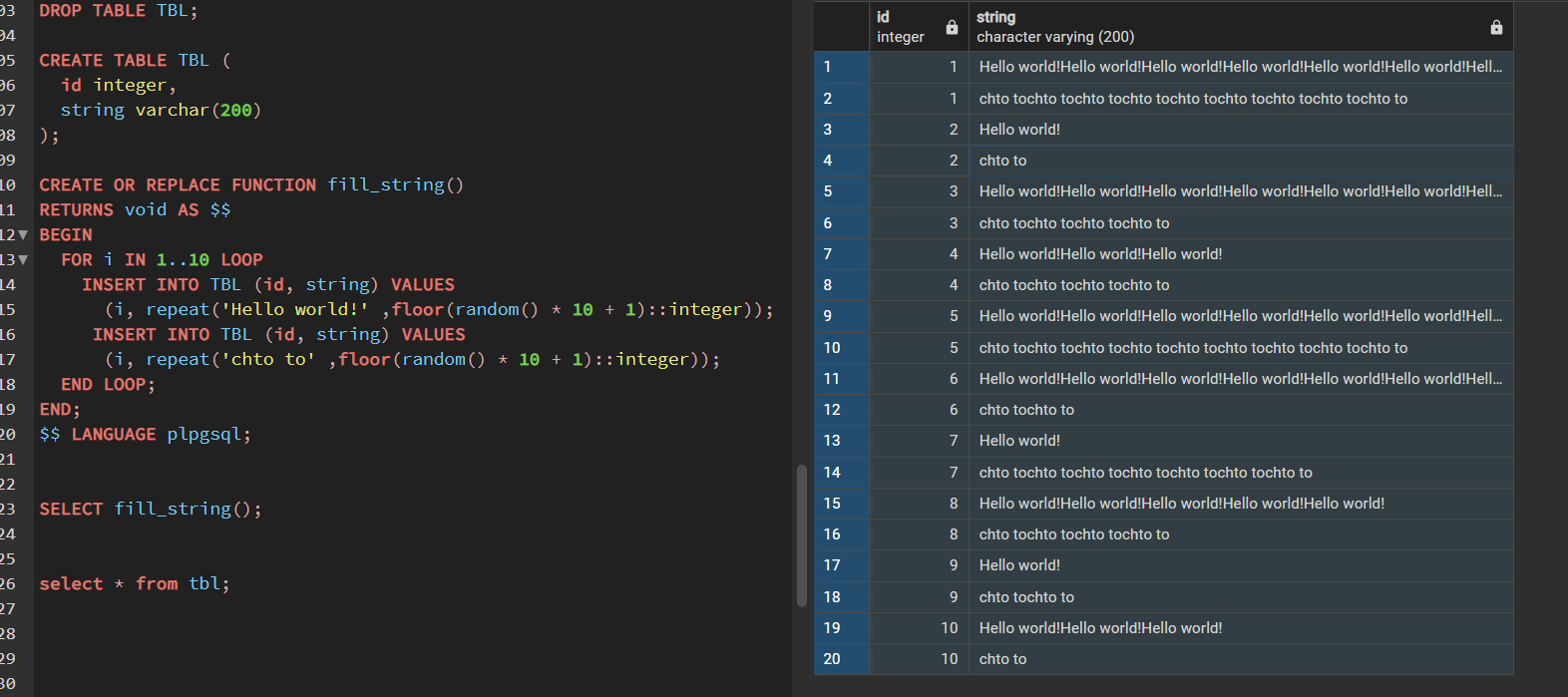
(length(string) - length(replace(string, 'Hello world!', ''))) / length('Hello world!') AS count\_per\_row,

((length(string) - length(replace(string, 'Hello world!', ''))) / length('Hello world!'))::float/summary\_count.summary\*100

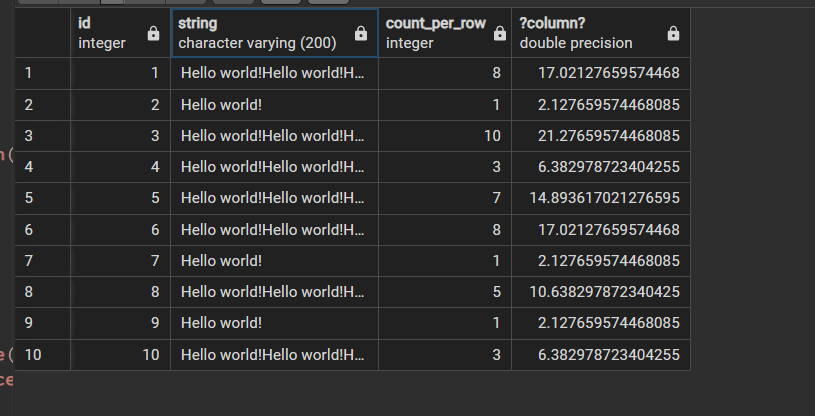
from tbl, summary\_count

where (length(string) - length(replace(string, 'Hello world!', ''))) / length('Hello world!') > 0

Вот так заполнил таблицу



Результат



WITH ranked\_data AS (

SELECT

name,

price,

ROW\_NUMBER() OVER(PARTITION BY name ORDER BY date DESC) AS rank

FROM TBL

)

SELECT

name AS "Валюта",

MAX(CASE WHEN rank = 1 THEN price END) AS "КУРС последний",

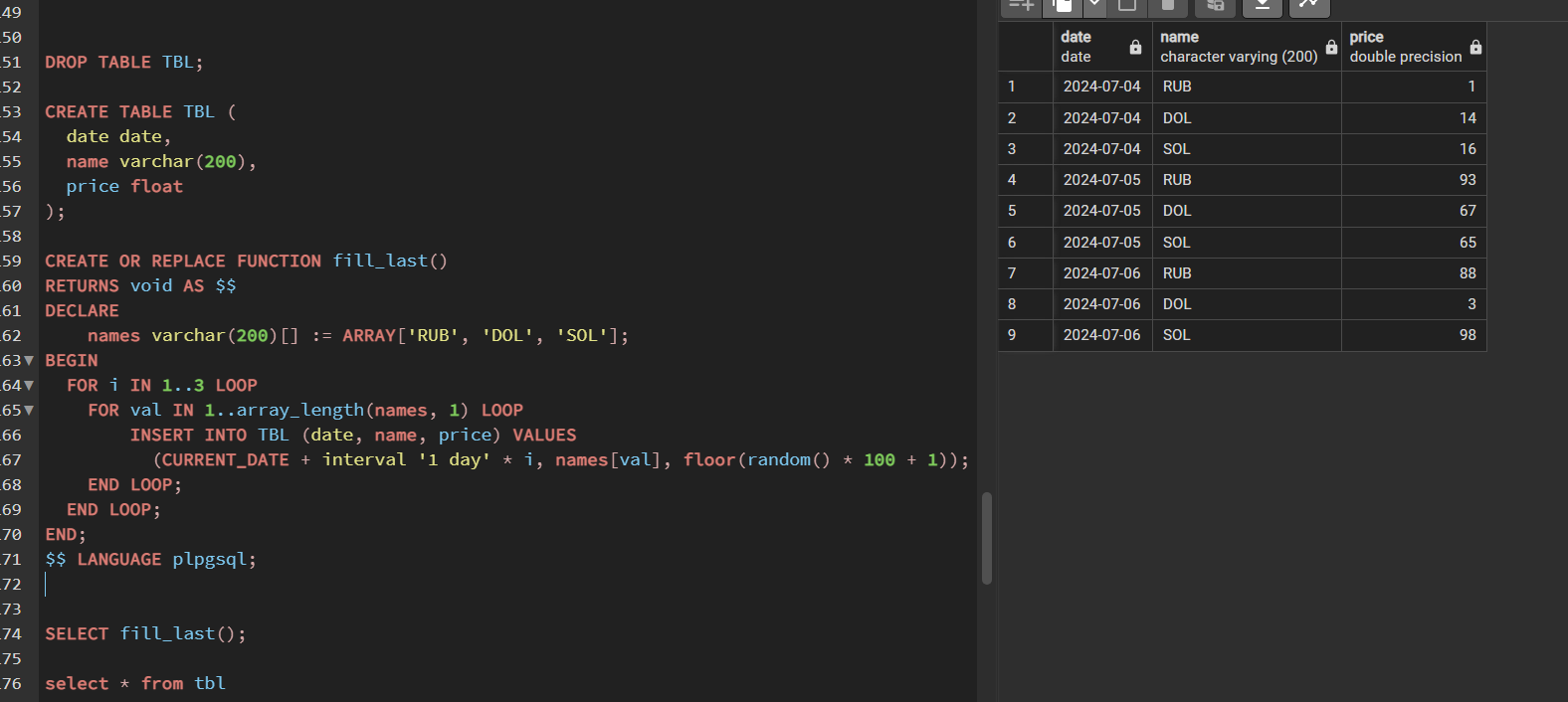
MAX(CASE WHEN rank = 2 THEN price END) AS "КУРС Предпоследний"

FROM ranked\_data

WHERE rank <= 2

GROUP BY name;

Заполнил таблицу



Результат

