Учреждение образования

«Белорусский государственный университет

информатики и радиоэлектронике»

Кафедра инженерной психологии и эргономики

Отчет по

ЛАБОРАТОРНОЙ РАБОТЕ №2

Разработка и реализация классов. Работа с данными с использованием Pandas

Подготовил:

Студент гр. 010101

Листванович А.А.

Проверила:

Василькова А.Н.

Минск 2023

Задача №1 – Вариант 5

Условие:

Напишите программу с классом Math. Создайте два атрибута — a и b. Напишите методы addition — сложение, multiplication — умножение, division — деление, subtraction — вычитание. При передаче в методы параметров a и b с ними нужно производить соответствующие действия и печатать ответ.

Решение:

def math\_operations() -> None:

class Math:

a: float

b: float

def addition(self) -> float:

return self.a + self.b

def subtraction(self) -> float:

return self.a - self.b

def multiplication(self) -> float:

return self.a \* self.b

def division(self) -> float:

return self.a / self.b

obj = Math()

dict\_of\_operations = {

'+': obj.addition,

'-': obj.subtraction,

'\*': obj.multiplication,

'/': obj.division

}

input\_str = ''

while input\_str != 'q':

input\_str = input('Выберите операцию: "+" "-" "\*" "/" "q". Введите "q", если хотите выйти:\n').lower().strip()

if input\_str == 'q':

print("Работа с программой завершена!")

break

if input\_str not in dict\_of\_operations.keys():

print('Ошибка!\nВыберите корректную операцию.')

continue

nums = 0

while nums != 2:

num\_str = input("Введите два числа через пробел:\n")

nums = len(num\_str.split())

if nums != 2:

print('Должно быть введено два числа!')

continue

obj.a, obj.b = [float(n) for n in num\_str.split()]

try:

print("Результат: ", dict\_of\_operations[input\_str]())

except ZeroDivisionError:

print('Не смогу я на нолик поделить, извини...')

Результат работы программы отображен на рисунке 1.

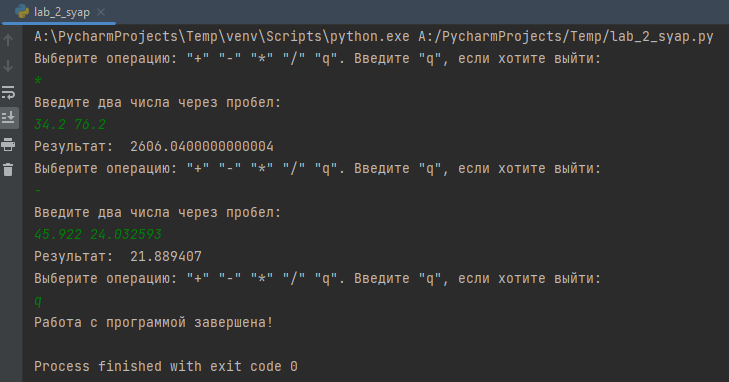


Рисунок 1 – Результат первой задачи

Задача №2

Условие:

Реализовать средствами Python и Pandas возможности Excel-функций IF и VLOOKUP.

Решение:

def if\_practice() -> None:

print('IF ----- PRACTICE')

df = pd.DataFrame(

{'States': ['California', 'Florida', 'Montana', 'Colorado', 'Washington', 'Virginia'],

'Capitals': ['Sacramento', 'Tallahassee', 'Helena', 'Denver', 'Olympia', 'Richmond'],

'Population': [508529, 193551, 32315, 619968, 52555, 227032]}

)

df.to\_excel('./states\_directory/states.xlsx')

print("Был создан excel-file с информацией о штатах!")

print(df)

min\_population = int(input('Минимальное население?\n'))

filtered\_df = df.query(f'Population>{min\_population}')

print("Ваша выборка сохранена в отдельный файл")

filtered\_df.to\_excel(f'./states\_directory/states\_population({min\_population}).xlsx')

print(filtered\_df)

def vlookup\_practice() -> None:

print('VLOOKUP ----- PRACTICE')

df\_1 = pd.DataFrame(

{'States': ['California', 'Florida', 'Montana', 'Colorado', 'Washington', 'Virginia'],

'Capitals': ['Sacramento', 'Tallahassee', 'Helena', 'Denver', 'Olympia', 'Richmond'],

'Population': [508529, 193551, 32315, 619968, 52555, 227032],

'Popularity': [7, 10, 7, 8, 10, 7]}

)

df\_2 = pd.DataFrame(

{'Popularity': [10, 8, 7],

'IsExpensive': [True, True, False]}

)

result\_df = pd.merge(df\_1, df\_2, how='left', on='Popularity')

print(result\_df)

def pandas\_practice() -> None:

if\_practice()

vlookup\_practice()

Результат работы программы отображен на рисунке 2.

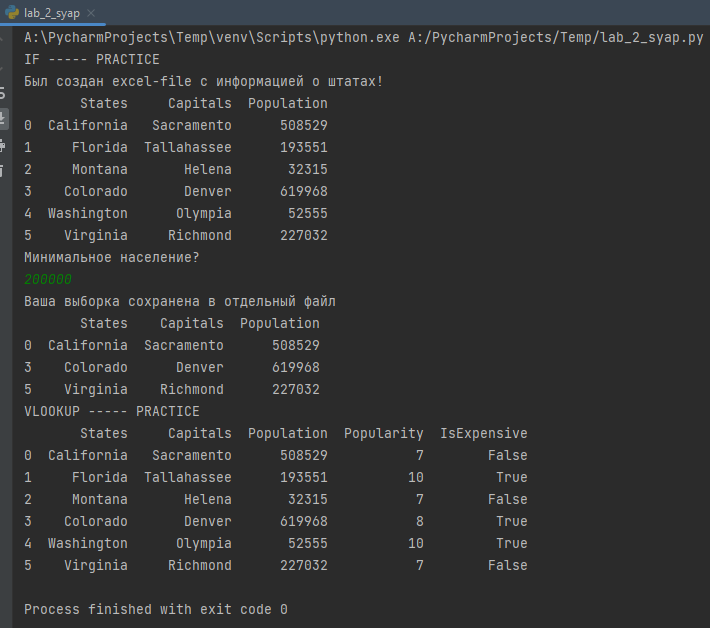


Рисунок 2 – Результат второй задачи