**Звіт про виконання практичних завдань до лекцій з курсу Технології програмування на мові Python**

Звіт до Теми №1

Функції та змінні

Під час виконання практичного завдання до Теми №1 було надано варіанти рішення до наступних задач:

**Перетворення рядка**

Необхідно рядок, що має вигляд "abcdefg123" перетворити наступним чином "321gfedcba", вважаючи сталою довжину рядку в 10 символів.

Хід виконання завдання:

В ПЗ пишем код:

Крок 1: Створюємо функцію 'revers', яка містить рядок 's' зі значенням "abcdefg123"

Крок 2: Перевертаємо рядок 's' за допомогою зрізу '[::-1]' і зберігаємо результат у змінну 'result'. (Перший порожній елемент означає "почни з першого символу" (або з кінця, бо ми йдемо у зворотному напрямку). Другий порожній елемент означає "йди до останнього символу" та -1 вказує на те, що ми рухаємося у зворотному порядку.

Крок 3: Функція повертає перевернутий рядок, який зберігається в 'result'

Текст програми:

|  |
| --- |
| def revers():      s ="abcdefg123"      result = s[::-1]      return result  print (revers()) |

При компіляції він виводить:



Посилання на github: https://github.com/Gendalfx/TP-KB-Andrii-Dukhvin

Знімок екрану з посилання на github:

**Виконати тестування функцій, що працюють з рядками: strip(), capitalize(), title(), upper(), lower().**

Текст програми:

def test\_string\_functions():

    s = "   hello world! welcome to python.   "

    # strip() - Видаляє пробіли з обох боків рядка

    stripped = s.strip()

    print(f"strip():'{stripped}'")

   # capitalize() - Робить першу літеру великою, а інші маленькими

    capitalized = stripped.capitalize()

    print(f"capitalize():'{capitalized}'")

    # title() - Робить першу літеру кожного слова великою

    titled = s.title()

    print(f"title():'{titled}'")

    # upper() - Перетворює всі символи в рядку на великі

    uppercased = s.upper()

    print(f"upper():'{uppercased}'")

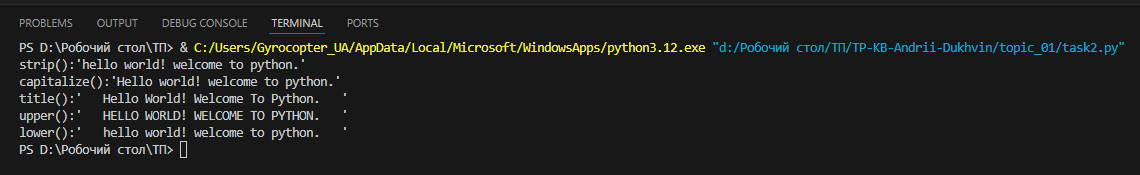
    # lower() - Перетворює всі символи в рядку на маленькі

    lowercased = s.lower()

    print(f"lower():'{lowercased}'")

test\_string\_functions()

При компіляції він виводить:



**Написати функцію пошуку дискримінанту квадратного рівняння.**

Формула дискримінанту: D=b^2−4ac

Текст програми:

def find\_discriminant(a, b, c):

    # Обчислюємо дискримінант

    discriminant = b\*\*2 - 4\*a\*c

    return discriminant

# Приклад використання

a = float(input("Введіть коефіцієнт a: "))

b = float(input("Введіть коефіцієнт b: "))

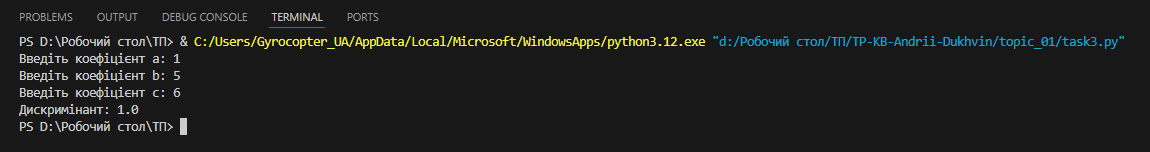
c = float(input("Введіть коефіцієнт c: "))

# Виклик функції та виведення результату

D = find\_discriminant(a, b, c)

print(f"Дискримінант: {D}")

При компіляції він виводить:



Перевірка:

