**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:

**1.Перетворення рядка**

Необхідно рядок, що має вигляд "qsdafw12" перетворити наступним чином "21qwfadsq", вважаючи сталою довжину рядку в 10 символів.

Хід виконання завдання:

Цей код виконує операцію перевертання рядка. Ось пояснення крок за кроком:

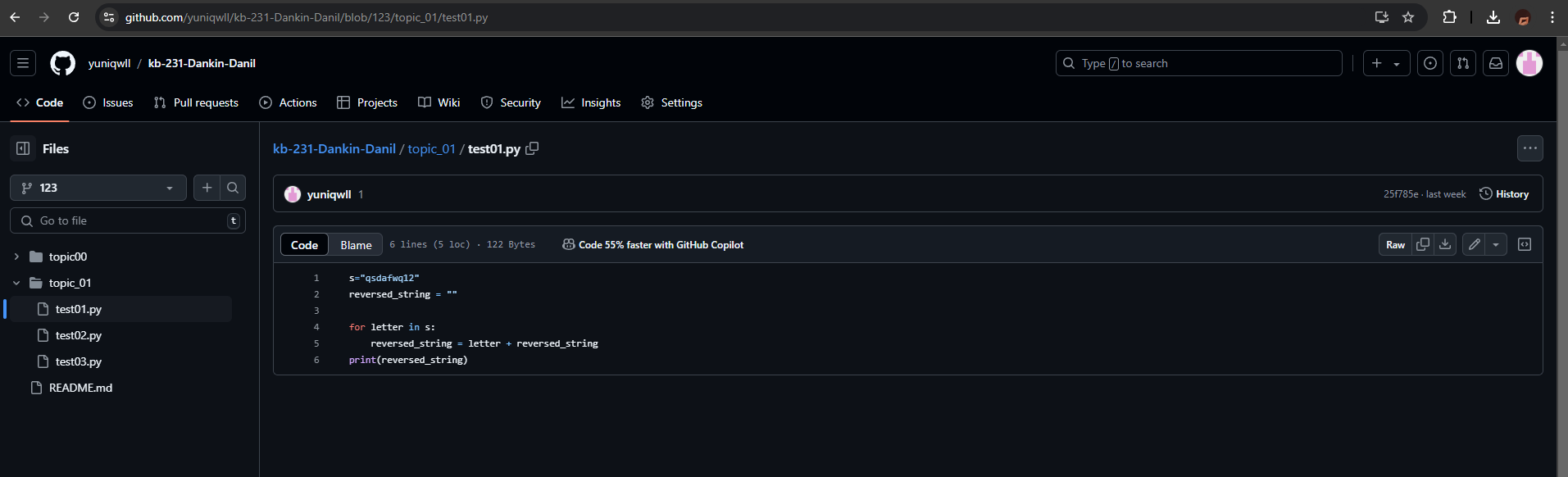
1. Початковий рядок s дорівнює "qsdafwq12"
2. Порожній рядок reversed\_string створюється для збереження результату.
3. Для кожної букви (або символа) в рядку s виконується наступне: Символ додається на початок рядка reversed\_string. Це означає, що кожен новий символ стає перед попередніми, поступово формуючи рядок у зворотному порядку.
4. Робота циклу:На першому кроці циклу змінна letter дорівнює "q", і значення reversed\_string стане "q".

На наступному кроці змінна letter дорівнюватиме "s", і рядок reversed\_string стане "sq". Цей процес продовжується для кожного символу.

Текст програми:

|  |
| --- |
| s="qsdafwq12"  reversed\_string = ""  for letter in s:      reversed\_string = letter + reversed\_string  print(reversed\_string) |

Посилання на github: https://github.com/yuniqwll/kb-231-Dankin-Danil/blob/123/topic\_01/test01.py

Знімок екрану з посилання на github: 

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

Хід виконання завдання:

1. Початковий рядок s дорівнює "Привіт світ ! " (з пробілом на початку та в кінці).
2. Метод strip() видаляє всі зайві пробіли на початку та в кінці рядка.
3. Метод capitalize() робить першу літеру всього рядка великою, а всі інші – малими.
4. Метод title() перетворює кожне слово в рядку так, що перша літера кожного слова стає великою, а інші – малими.
5. Метод upper() переводить всі літери в рядку в верхній регістр (великі літери).
6. Метод lower() переводить всі літери в нижній регістр (малі літери).

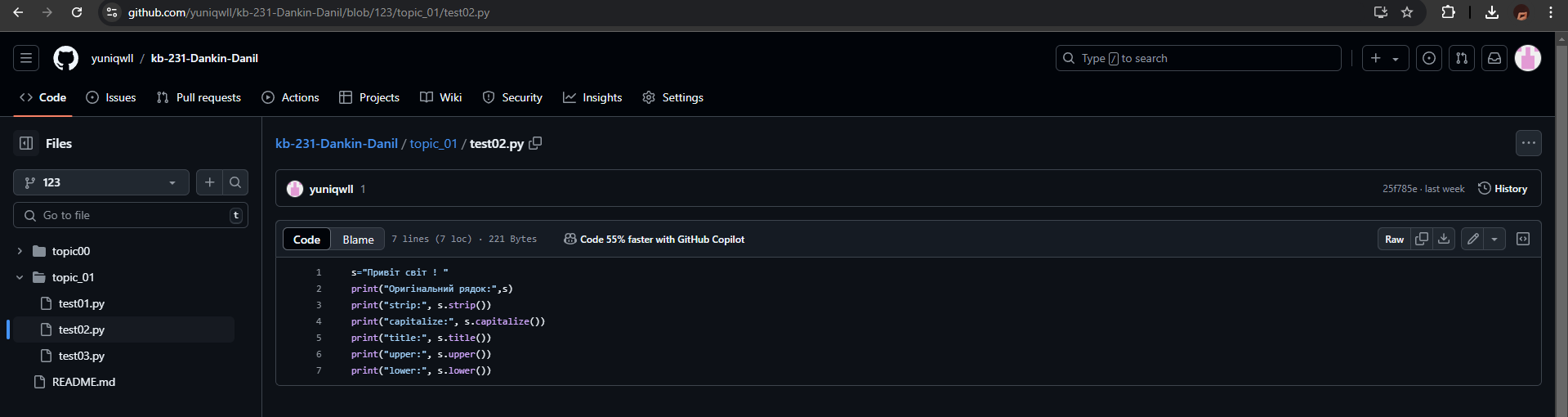
Цей код демонструє роботу з рядками за допомогою кількох вбудованих методів Python для зміни регістру літер і роботи з пробілами.

Текст програми: Текст програми:

|  |
| --- |
| s="Привіт світ ! "  print("Оригінальний рядок:",s)  print("strip:", s.strip())  print("capitalize:", s.capitalize())  print("title:", s.title())  print("upper:", s.upper())  print("lower:", s.lower()) |

Посилання на github: <https://github.com/yuniqwll/kb-231-Dankin-Danil/blob/123/topic_01/test02.py>

Знімок екрану з посилання на github:



**3. Написати функцію пошуку дискримінанту квадратного рівняння.**

Хід виконання завдання:

1) **Оголошення функції findD(a, b, c)**:

Функція приймає три аргументи: a, b та c, які є коефіцієнтами квадратного рівняння.

Усередині функції виконується обчислення дискримінанта за формулою D=b2−4acD = b^2 - 4acD=b2−4ac.

Оператор return повертає значення дискримінанта

2) **Ведення значень**:

За допомогою функції input() програма запитує у користувача значення коефіцієнтів:

a — для коефіцієнта при x^2,

b — для коефіцієнта при x,

c — вільного члена.

Введені значення перетворюються на цілі числа за допомогою функції int()

3) **Обчислення дискримінанта**:

Функція findD(a, b, c) викликається із переданими значеннями коефіцієнтів, і обчислюється дискримінант

Обчислене значення дискримінанта виводиться на екран за допомогою функції print().

Текст програми:

def findD(a,b,c):

    return b\*\*2-4\*a\*c

a=int(input("Введіть коофіцієнт a: "))

b=int(input("Введіть коофіцієнт b: "))

c=int(input("Введіть коофіцієнт c: "))

print("Дискримінант:", findD(a, b ,c))

Посилання на github: <https://github.com/yuniqwll/kb-231-Dankin-Danil/blob/123/topic_01/test03.py>

Знімок екрану з посилання на github: 