**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:**

**1.** Задаю рядок text = "abcdefg123".

**2.** Використовую зріз [::-1], щоб перевернути весь рядок і зберегти у змінній reversed\_text.

**3.** Виводжу оригінальний рядок через print(text).

**4.** Виводжу перевернутий рядок через print(reversed\_text).

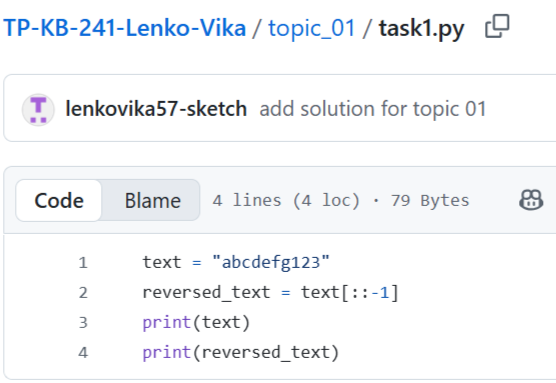
**Текст програми:**

|  |
| --- |
| text = "abcdefg123"  reversed\_text = text[::-1]  print(text)  print(reversed\_text) |

**Посилання на github:**

<https://github.com/lenkovika57-sketch/TP-KB-241-Lenko-Vika/blob/main/topic_01/task1.py>

**Знімок екрану з посилання на github:**



**Тестування функцій**

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

**Хід виконання завдання:**

**1.** Задаю рядок text = " heLLo, wORLD! " з пробілами на початку і в кінці.

**2.** Використовую функцію strip(), щоб прибрати пробіли спереду і ззаду.

**3.** Використовую capitalize(), щоб першу букву рядка зробити великою, а решту - маленькими.

**4.** Використовую title(), щоб зробити великою першу букву кожного слова.

**5.** Використовую upper(), щоб перетворити всі літери на великі.

**6.** Використовую lower(), щоб перетворити всі літери на маленькі.

**7.** Виводжу результати кожної функції через print().

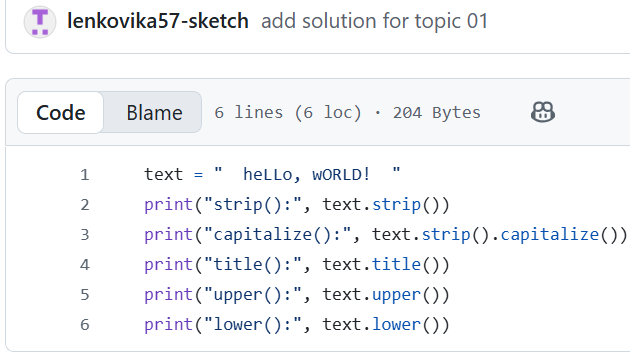
**Текст програми:**

|  |
| --- |
| text = " heLLo, wORLD! "  print("strip():", text.strip())  print("capitalize():", text.strip().capitalize())  print("title():", text.title())  print("upper():", text.upper())  print("lower():", text.lower()) |

**Посилання на github:**

<https://github.com/lenkovika57-sketch/TP-KB-241-Lenko-Vika/blob/main/topic_01/task2.py>

**Знімок екрану з посилання на github:**



**Написання функції**

Необхідно написати функцію пошуку дискримінанту квадратного рівняння.

**Хід виконання завдання:**

**1.** Створюю функцію discr(a, b, c), яка обчислює дискримінант за формулою D = b² - 4ac.

**2.** Через input() отримую значення коефіцієнтів a, b і c від користувача.

**3.** Викликаю функцію discr(a, b, c) і зберігаю результат у змінній D.

**4.** Виводжу результат на екран через print().

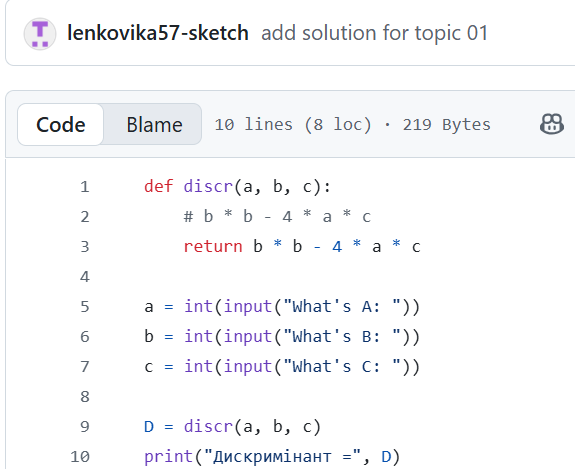
**Текст програми:**

|  |
| --- |
| def discr(a, b, c):  # b \* b - 4 \* a \* c  return b \* b - 4 \* a \* c  a = int(input("What's A: "))  b = int(input("What's B: "))  c = int(input("What's C: "))  D = discr(a, b, c)  print("Дискримінант =", D) |

**Посилання на github:**

<https://github.com/lenkovika57-sketch/TP-KB-241-Lenko-Vika/blob/main/topic_01/task3.py>

**Знімок екрану з посилання на github:**



Звіт до Теми №2

[Умовний перехід](https://eln.stu.cn.ua/course/section.php?id=98202)

Під час виконання практичного завдання до Теми №2 було надано варіанти рішення до наступних задач:

**Написання функції пошуку**

Написати функцію пошуку коренів квадратного рівняння використовуючи функцію розрахунку дискримінанту з попередньої теми та умовні переходи.

**Хід виконання завдання:**

**1.** Імпортую модуль math для кореня.

**2.** Створюю функцію для обчислення дискримінанта D = b^2 - 4ac.

**3.** Створюю функцію, яка обчислює D, аналізує D, виводить результат.

**4.** Отримую коефіцієнти a, b, c від користувача.

**5.** Викликаю функцію quadratic\_roots(a, b, c).

**Текст програми:**

import math

def discr(a, b, c):

    return b \* b - 4 \* a \* c

def quadratic\_roots(a, b, c):

    D = discr(a, b, c)

    print("Дискримінант =", D)

    if D < 0:

        print("Коренів немає")

    elif D == 0:

        x = -b / (2 \* a)

        print("Один корінь: x =", x)

    else:

        x1 = (-b + math.sqrt(D)) / (2 \* a)

        x2 = (-b - math.sqrt(D)) / (2 \* a)

        print("Два корені: x1 =", x1, "x2 =", x2)

a = int(input("Введіть A: "))

b = int(input("Введіть B: "))

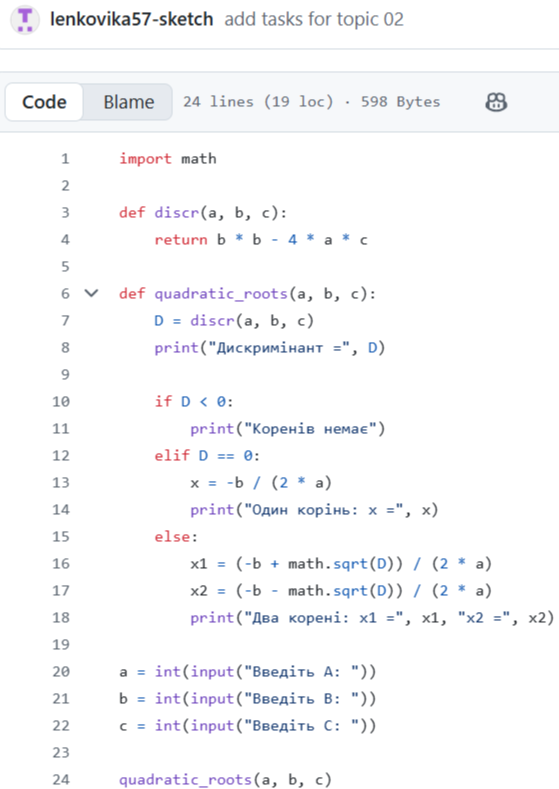
c = int(input("Введіть C: "))

quadratic\_roots(a, b, c)

**Посилання на github:**

<https://github.com/lenkovika57-sketch/TP-KB-241-Lenko-Vika/blob/main/topic_02/task1.py>

**Знімок екрану з посилання на github:**



**Написання програми використовуючи** **if else**

Написати програму калькулятор використовуючи **if else** конструкцію. Кожна операція має бути виконана в окремій функції.

**Хід виконання завдання:**

**1.** Створюю функцію, яка приймає два числа (a, b) та дію (sign).

**2.** За допомогою if/elif перевіряю sign (+, -, \*, /) та повертаю результат операції.

**3.** У випадку ділення (/) перевіряю, чи b не дорівнює 0, інакше повертаю помилку.

**4.** Отримую числа a, b (як float) та дію sign (як string) від користувача.

**5.** Викликаю функцію calculator(a, b, sign) і виводжу результат.

**Текст програми:**

def calculator(a, b, sign):

    if sign == "+":

        return a + b

    elif sign == "-":

        return a - b

    elif sign == "\*":

        return a \* b

    elif sign == "/":

        if b != 0:

            return a / b

        else:

            return "Ділити на 0 неможливо"

    else:

        return "Некоректна дія"

a = float(input("Перше число: "))

b = float(input("Друге число: "))

sign = input("Дія (+ - \* /): ")

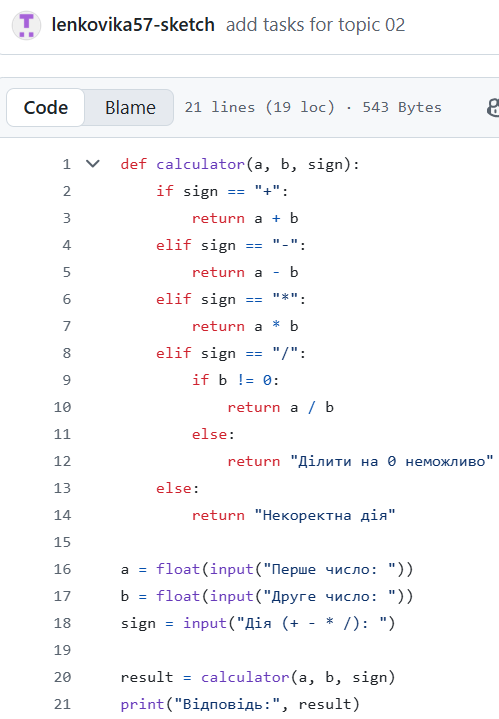
result = calculator(a, b, sign)

print("Відповідь:", result)

**Посилання на github:**

https://github.com/lenkovika57-sketch/TP-KB-241-Lenko-Vika/blob/main/topic\_02/task2.py

**Знімок екрану з посилання на github:**



**Написання програми використовуючи match**

Написати програму калькулятор використовуючи **match** конструкцію. Кожна операція має бути виконана в окремій функції.

**Хід виконання завдання:**

**1.** Створюю функцію, яка приймає a, b та дію sign.

**2.** Використовую match sign: для виконання операцій (+, -, \*).

**3.** У випадку / перевіряю, чи b \neq 0. Якщо b=0, повертаю помилку.

**4.** Використовую case \_ для обробки некоректної дії.

**5.** Отримую a, b, sign від користувача та виводжу результат функції.

**Текст програми:**

def calculator(a, b, sign):

    match sign:

        case "+":

            return a + b

        case "-":

            return a - b

        case "\*":

            return a \* b

        case "/":

            if b != 0:

                return a / b

            else:

                return "Ділити на 0 неможливо"

        case \_:

            return "Некоректна дія"

a = float(input("Перше число: "))

b = float(input("Друге число: "))

sign = input("Дія (+ - \* /): ")

result = calculator(a, b, sign)

print("Відповідь:", result)

**Посилання на github:**

https://github.com/lenkovika57-sketch/TP-KB-241-Lenko-Vika/blob/main/topic\_02/task3.py

**Знімок екрану з посилання на github:**

