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

Звіт до Теми №1

Функції та змінні

﻿﻿Завдання

1. Обернути рядок в зворотному порядку.
2. ﻿﻿Виконати тестування функцій, що працюють з рядками: strip(), capitalize(), title(), upper(), lower() ﻿﻿﻿
3. Написати функцію пошуку дискримінанту квадратного рівняння.

Хід виконання завдання

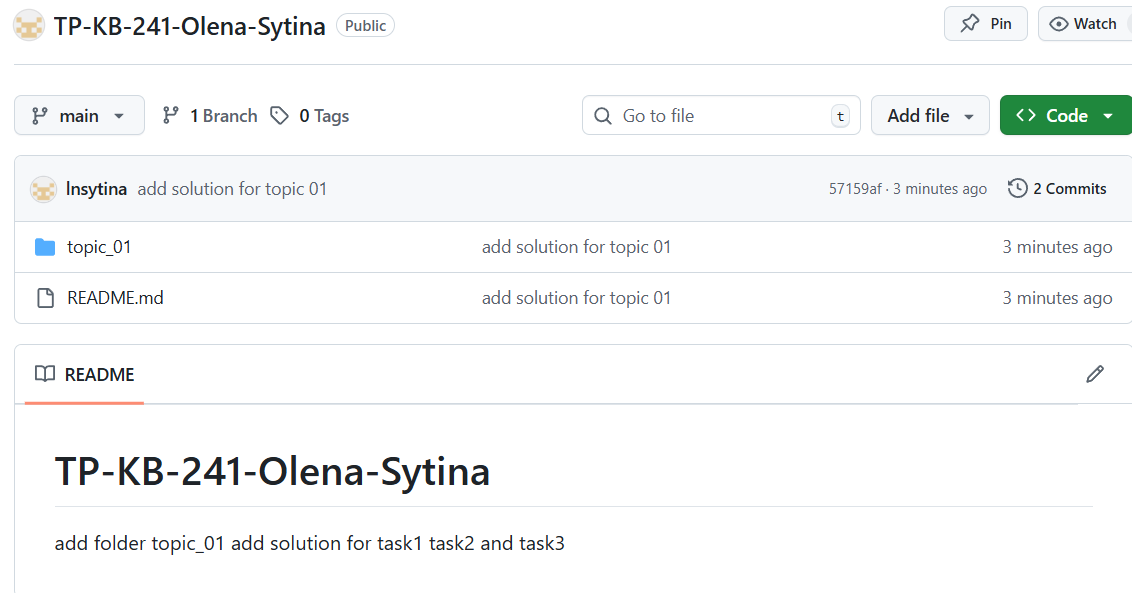
Текст програми:

|  |
| --- |
| #task1  def reverse\_string(text):  return text[::-1]  s = input("Введіть рядок: ")  print("Обернений рядок:", reverse\_string(s))  #task2  a = "hello world"  print(a.strip())  print(a.capitalize())  print(a.title())  print(a.upper())  print(a.lower())    #task3  import math  print("Введіть коефіцієнти квадратного рівняння ax^2 + bx + c = 0: ")  a = int(input("a = "))  b = int(input("b = "))  c = int(input("c = "))    D = b \*\* 2 - 4 \* a \* c  print(D)  if D > 0:  x1 = (-b + math.sqrt(D)) / (2 \* a)  x2 = (-b - math.sqrt(D)) / (2 \* a)  print(x1, x2)  elif D == 0:  x = -b / (2 \* a)  print(x)  else:  print("Коренів не має") |

Посилання на github:

<https://github.com/lnsytina/TP-KB-241-Olena-Sytina>

Знімок екрану з посилання на github:



Звіт до Теми №2

Умовний перехід

﻿﻿Завдання

1. Написати функцію пошуку коренів квадратного рівняння використовуючи функцію розрахунку дискримінанту з попередньої теми та умовні переходи.
2. Написати програму калькулятор використовуючи if else конструкцію. Кожна операція має бути виконана в окремій функції.
3. Написати програму калькулятор використовуючи match конструкцію. Кожна операція має бути виконана в окремій функції.

Текст програми:

|  |
| --- |
| #task1  import math  print("Введіть коефіцієнти квадратного рівняння ax^2 + bx + c = 0: ")  a = int(input("a = "))  b = int(input("b = "))  c = int(input("c = "))    D = b \*\* 2 - 4 \* a \* c  print(D)  if D > 0:  x1 = (-b + math.sqrt(D)) / (2 \* a)  x2 = (-b - math.sqrt(D)) / (2 \* a)  print(x1, x2)  elif D == 0:  x = -b / (2 \* a)  print(x)  else:  print("Коренів не має")  #task2  def add(a, b):  return a + b  def minus(a, b):  return a - b  def mult(a, b):  return a \* b  def div(a, b):  if b != 0:  return a / b  else:  return "Ділення на нуль!"  print("Оберіть операцію: +, -, \*, /")  s = input("Введіть знак операції: ")  a = float(input("Перше число: "))  b = float(input("Друге число: "))  if s == "+":  print("Результат:", add(a, b))  elif s == "-":  print("Результат:", minus(a, b))  elif s == "\*":  print("Результат:", mult(a, b))  elif s == "/":  print("Результат:", div(a, b))  else:  print("Невірний знак!")  #task3  def add(a, b):  return a + b  def minus(a, b):  return a - b  def mult(a, b):  return a \* b  def div(a, b):  if b != 0:  return a / b  else:  return "Ділення на нуль!"  print("Оберіть операцію: +, -, \*, /")  s = input("Введіть знак операції: ")  a = float(input("Перше число: "))  b = float(input("Друге число: "))  match s:  case "+":  print("Результат:", add(a, b))  case "-":  print("Результат:", minus(a, b))  case "\*":  print("Результат:", mult(a, b))  case "/":  print("Результат:", div(a, b))  case \_:  print("Невірна операція!") |

Звіт до Теми №3

Цикли

﻿﻿Завдання

1) Написати програму калькулятор з постійними запитами на введення нових даних та операцій. За основу взяти програму калькулятор з попередньої теми. Реалізувати механізм завершення програми після отримання відповідної команди.

2) Написати програму тестування функцій списків таких як: extend(), append(), insert(id, val), remove(val), clear(), sort(), reverse(), copy()

3) Написати програму тестування функцій словників таких як: update(), del(), clear(), keys(), values(), items()

4) Маючи відсортований список, написати функцію пошуку позиції для вставки нового елементу в список.

Текст програми:

|  |
| --- |
| #task1  def add(a, b):  return a + b  def minus(a, b):  return a - b  def mult(a, b):  return a \* b  def div(a, b):  if b != 0:  return a / b  else:  return "Ділення на нуль!"    while True:  s = input("Введіть знак операції: або якщо хочете закрити калькулятор введіть exit ")    if s == "exit":  print("Програма завершина")  break  else:  a = float(input("Перше число: "))  b = float(input("Друге число: "))  match s:  case "+":  print("Результат:", add(a, b))  case "-":  print("Результат:", minus(a, b))  case "\*":  print("Результат:", mult(a, b))  case "/":  print("Результат:", div(a, b))  case \_:  print("Невірна операція!")  #task2  list = [3, 1, 4]  print("Початковий список:", list)  list.append(5)  print("append(5):", list)  list.extend([7, 9])  print("extend([7, 9]):", list)  list.insert(1, 10)  print("insert(1, 10):", list)  list.remove(4)  print("remove(4):", list)  list.sort()  print("sort():", list)  list.reverse()  print("reverse():", list)  copy\_list = list.copy()  print("copy():", copy\_list)  list.clear()  print("clear():", list)  #task3  book = {"1": "Hello", "2": "World"}  print("Cловник:", book)  book.update({"3": "Good"})  print("update({'3': 'Good'}):", book)  print("keys():", book.keys())  print("values():", book.values())  print("items():", book.items())  del book["2"]  print("del 2:", book)  book.clear()  print("clear():", book)  #task4  def find\_insert\_position(list, value):  for i in range(len(list)):  if value < list[i]:  return i  return len(list)  sorted\_list = [1, 3, 5, 7, 9]  print("Відсортований список:", sorted\_list)  new\_value = int(input("Введіть число для вставки: "))  position = find\_insert\_position(sorted\_list, new\_value)  print("Позиція для вставки:", position)  sorted\_list.insert(position, new\_value)  print("Новий список:", sorted\_list) |