**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. Відкрила ВСкод
2. Прочитала завдання
3. Виконала завдання
4. Затестила
5. Закинула в звіт

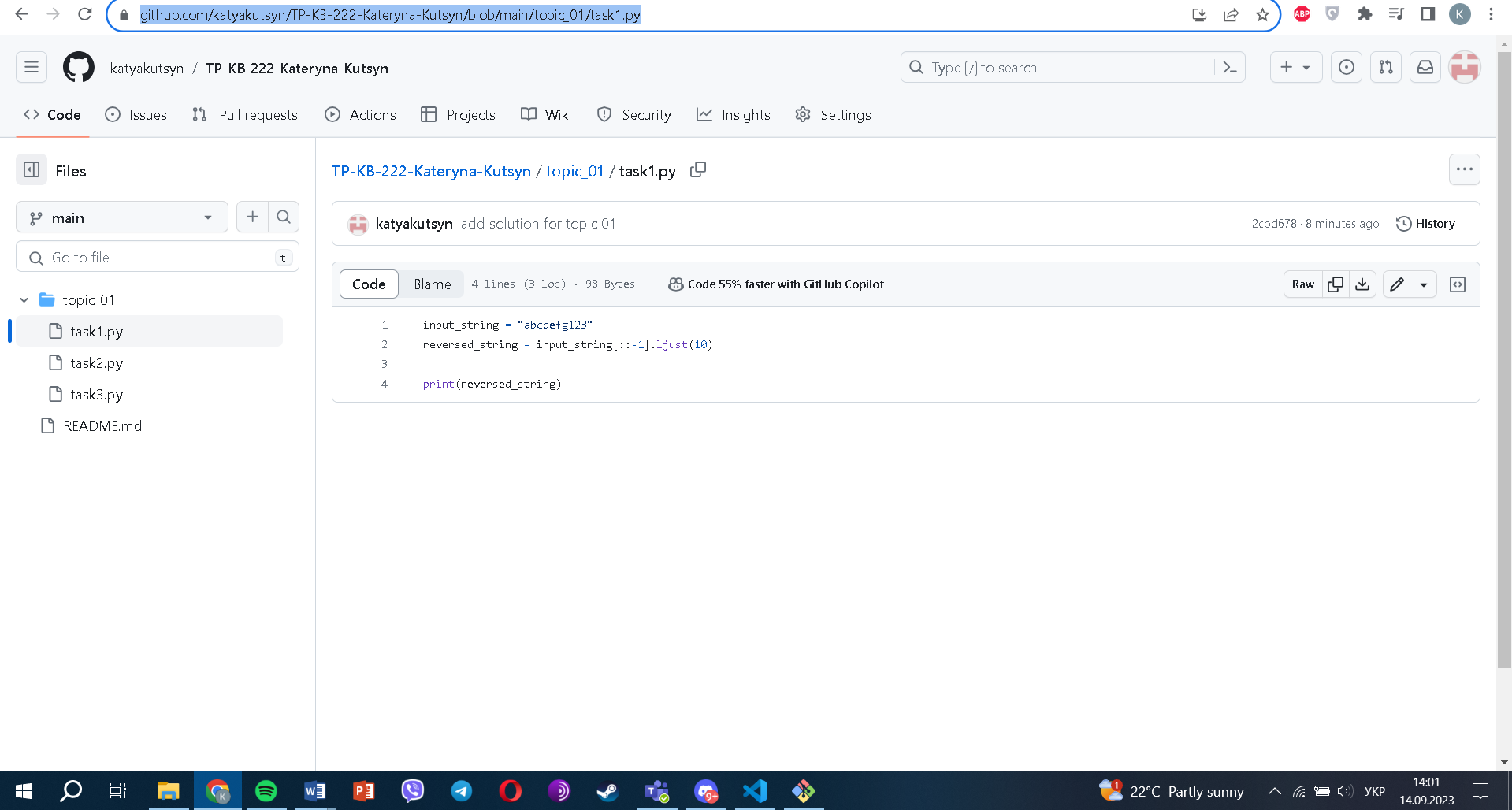
Текст програми:

|  |
| --- |
| input\_string = "abcdefg123"  reversed\_string = input\_string[::-1]  print(reversed\_string) |

Посилання на github:

<https://github.com/katyakutsyn/TP-KB-222-Kateryna-Kutsyn/blob/main/topic_01/task1.py>

Знімок екрану з посилання на github:



**Форматування рядка**

Виконайте певний тест для strip, capitalize, title, upper, lower, replace

Хід виконання завдання:

1. Відкрила ВСкод
2. Прочитала завдання
3. Виконала завдання
4. Затестила
5. Закинула в звіт

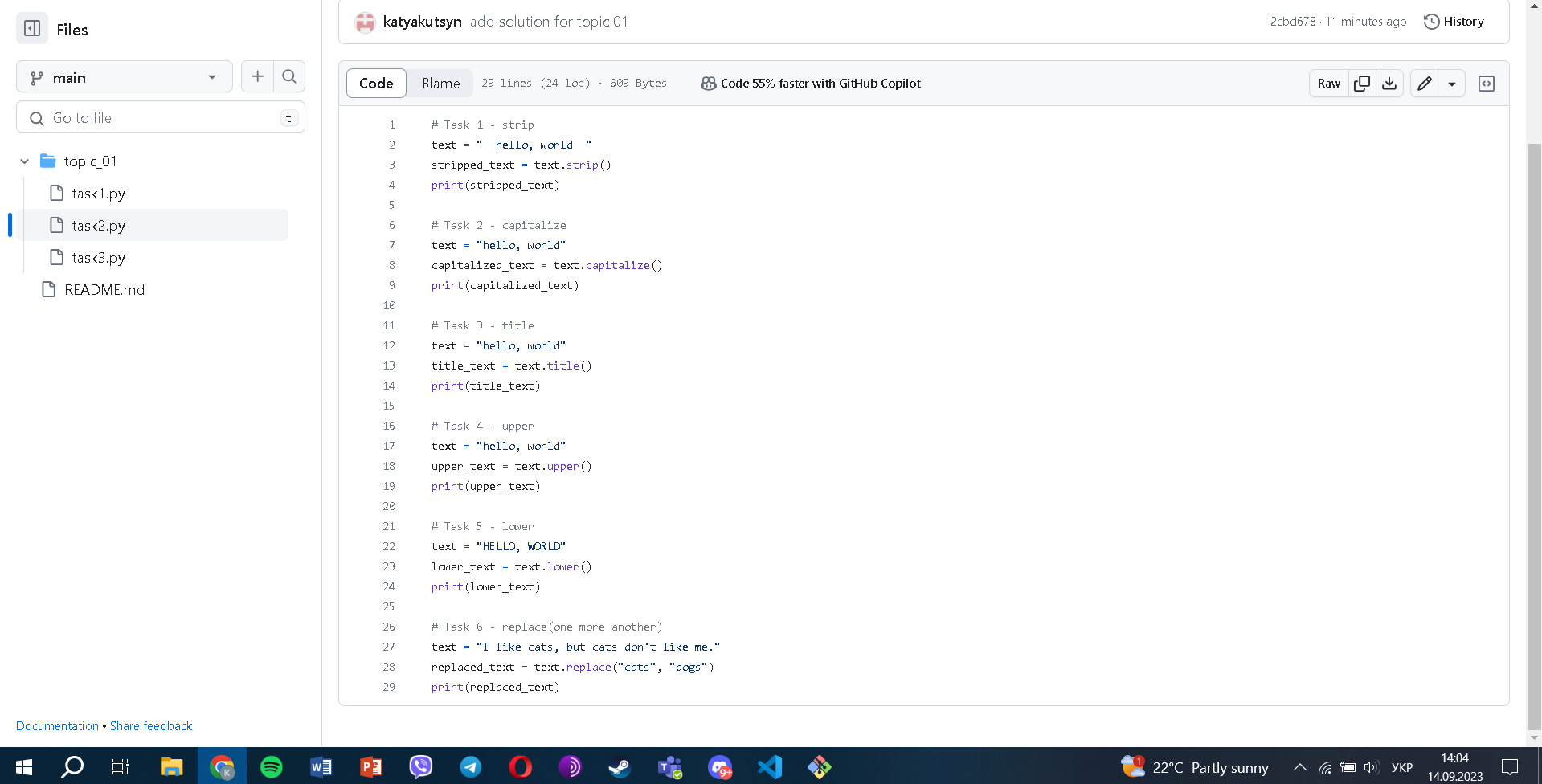
Текст програми:

|  |
| --- |
| # Task 1 - strip  text = " hello, world "  stripped\_text = text.strip()  print(stripped\_text)  # Task 2 - capitalize  text = "hello, world"  capitalized\_text = text.capitalize()  print(capitalized\_text)  # Task 3 - title  text = "hello, world"  title\_text = text.title()  print(title\_text)  # Task 4 - upper  text = "hello, world"  upper\_text = text.upper()  print(upper\_text)  # Task 5 - lower  text = "HELLO, WORLD"  lower\_text = text.lower()  print(lower\_text)  # Task 6 - replace(one more another)  text = "I like cats, but cats don't like me."  replaced\_text = text.replace("cats", "dogs")  print(replaced\_text) |

Посилання на github:

[https://github.com/katyakutsyn/TP-KB-222-Kateryna- Kutsyn/blob/main/topic\_01/task2.py](https://github.com/katyakutsyn/TP-KB-222-Kateryna-%20%20Kutsyn/blob/main/topic_01/task2.py)

Знімок екрану з посилання на github:



**Функція для знаходження дискримінанту**

Виконайте і напишіть функцію для знаходження дискримінанту

Хід виконання завдання:

1. Відкрила ВСкод
2. Прочитала завдання
3. Виконала завдання
4. Затестила
5. Закинула в звіт

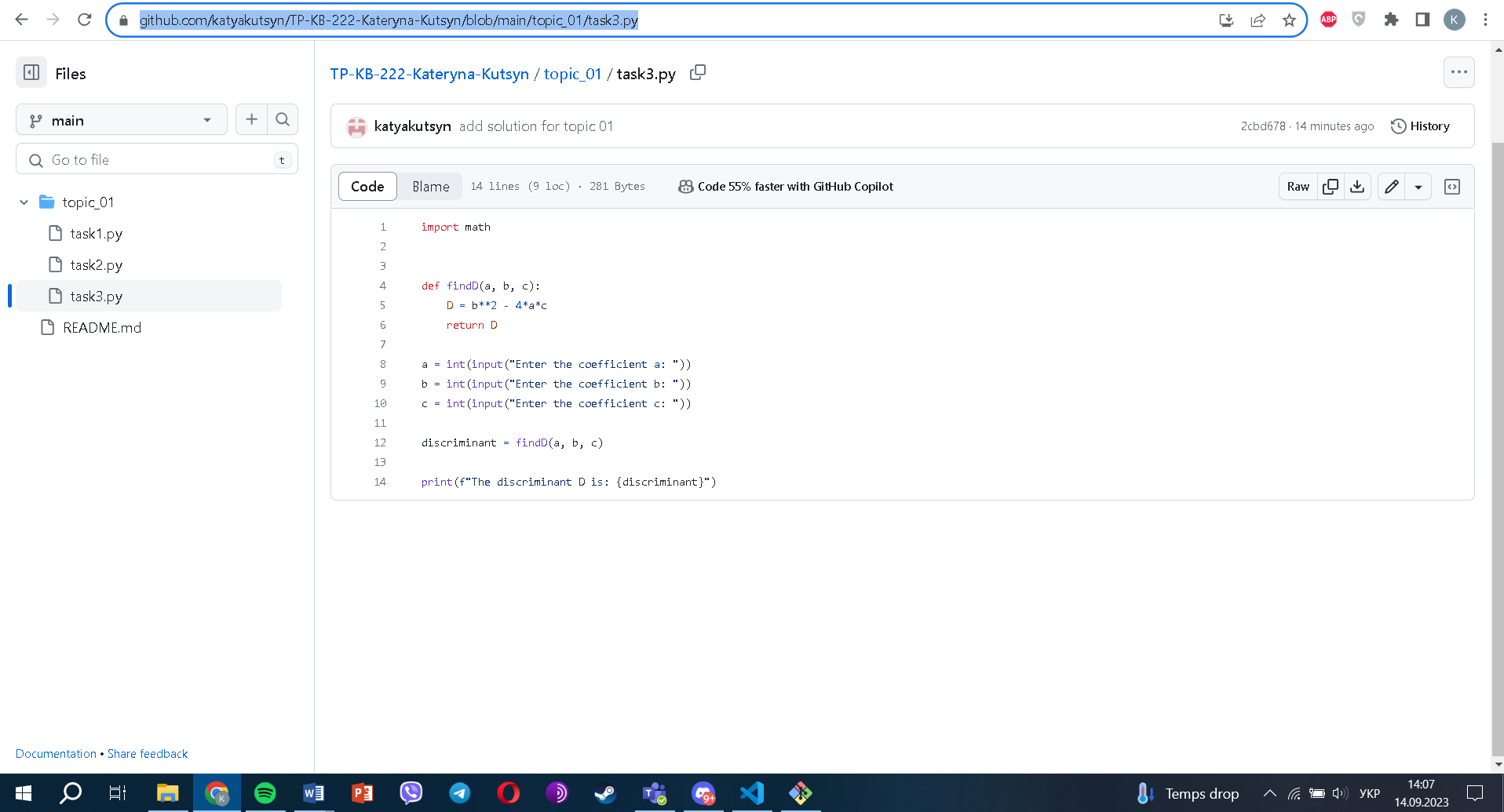
Текст програми:

|  |
| --- |
| import math  def findD(a, b, c):  D = b\*\*2 - 4\*a\*c  return D  a = int(input("Enter the coefficient a: "))  b = int(input("Enter the coefficient b: "))  c = int(input("Enter the coefficient c: "))  discriminant = findD(a, b, c)  print(f"The discriminant D is: {discriminant}") |

Посилання на github:

<https://github.com/katyakutsyn/TP-KB-222-Kateryna-Kutsyn/blob/main/topic_01/task3.py>

Знімок екрану з посилання на github:



Звіт до Теми №2

Функції та змінні

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

**Знаходження Дискримінанту**

rewrite function for quadratic equation with using conditional **for D**

D < 0

D == 0

D > 0

Хід виконання завдання:

1. Відкрила ВСкод
2. Прочитала завдання
3. Виконала завдання
4. Затестила
5. Закинула в звіт

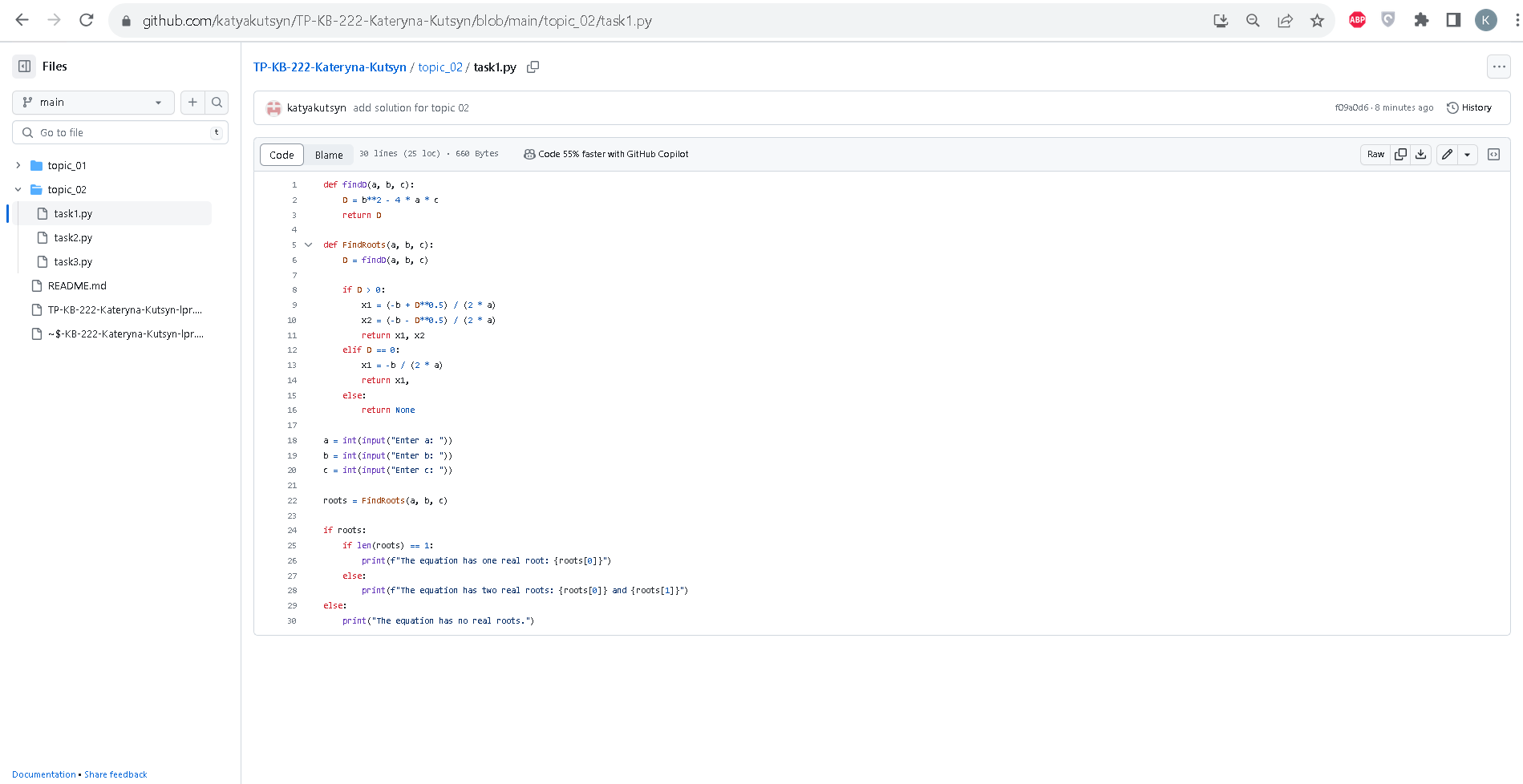
Текст програми:

|  |
| --- |
| def findD(a, b, c):  D = b\*\*2 - 4 \* a \* c  return D  def FindRoots(a, b, c):  D = findD(a, b, c)    if D > 0:  x1 = (-b + D\*\*0.5) / (2 \* a)  x2 = (-b - D\*\*0.5) / (2 \* a)  return x1, x2  elif D == 0:  x1 = -b / (2 \* a)  return x1,  else:  return None  a = int(input("Enter a: "))  b = int(input("Enter b: "))  c = int(input("Enter c: "))  roots = FindRoots(a, b, c)  if roots:  if len(roots) == 1:  print(f"The equation has one real root: {roots[0]}")  else:  print(f"The equation has two real roots: {roots[0]} and {roots[1]}")  else:  print("The equation has no real roots.") |

Посилання на github:

<https://github.com/katyakutsyn/TP-KB-222-Kateryna-Kutsyn/blob/main/topic_02/task1.py>

Знімок екрану з посилання на github:



**Створити калькулятор з if elif**

First using if elif. There are 2 values as input parameters and operation to do + - \* /

Хід виконання завдання:

1. Відкрила ВСкод
2. Прочитала завдання
3. Виконала завдання
4. Затестила
5. Закинула в звіт

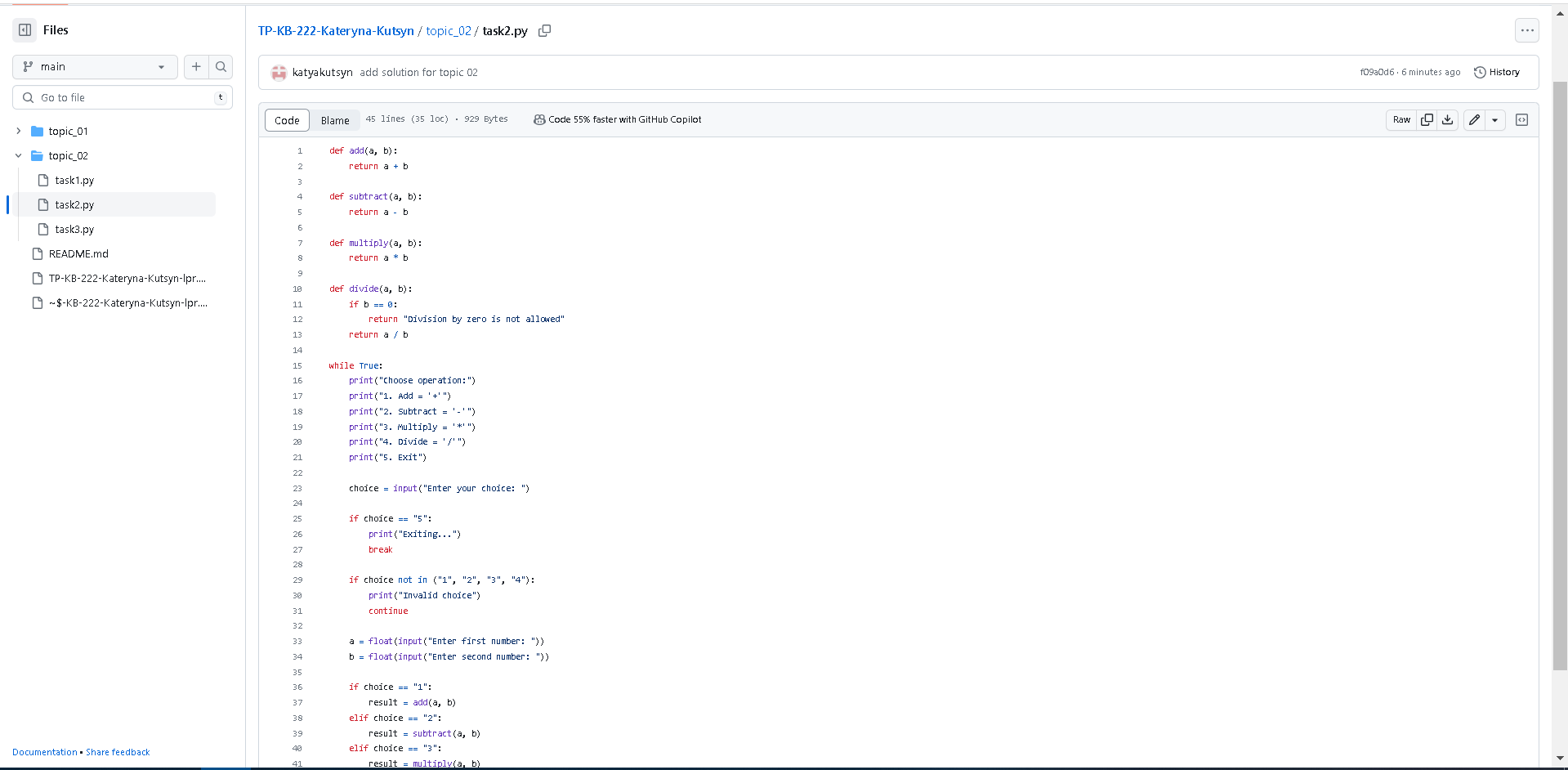
Текст програми:

|  |
| --- |
| def add(a, b):  return a + b  def subtract(a, b):  return a - b  def multiply(a, b):  return a \* b  def divide(a, b):  if b == 0:  return "Division by zero is not allowed"  return a / b  while True:  print("Choose operation:")  print("1. Add = '+'")  print("2. Subtract = '-'")  print("3. Multiply = '\*'")  print("4. Divide = '/'")  print("5. Exit")  choice = input("Enter your choice: ")  if choice == "5":  print("Exiting...")  break  if choice not in ("1", "2", "3", "4"):  print("Invalid choice")  continue  a = float(input("Enter first number: "))  b = float(input("Enter second number: "))  if choice == "1":  result = add(a, b)  elif choice == "2":  result = subtract(a, b)  elif choice == "3":  result = multiply(a, b)  elif choice == "4":  result = divide(a, b)  print("Result:", result) |

Посилання на github:

<https://github.com/katyakutsyn/TP-KB-222-Kateryna-Kutsyn/blob/main/topic_02/task2.py>

Знімок екрану з посилання на github:



**Створити калькулятор з** **match**

Second using match. There are 2 values as input parameters and operation to do + - \* /

Хід виконання завдання:

1. Відкрила ВСкод
2. Прочитала завдання
3. Виконала завдання
4. Затестила
5. Закинула в звіт

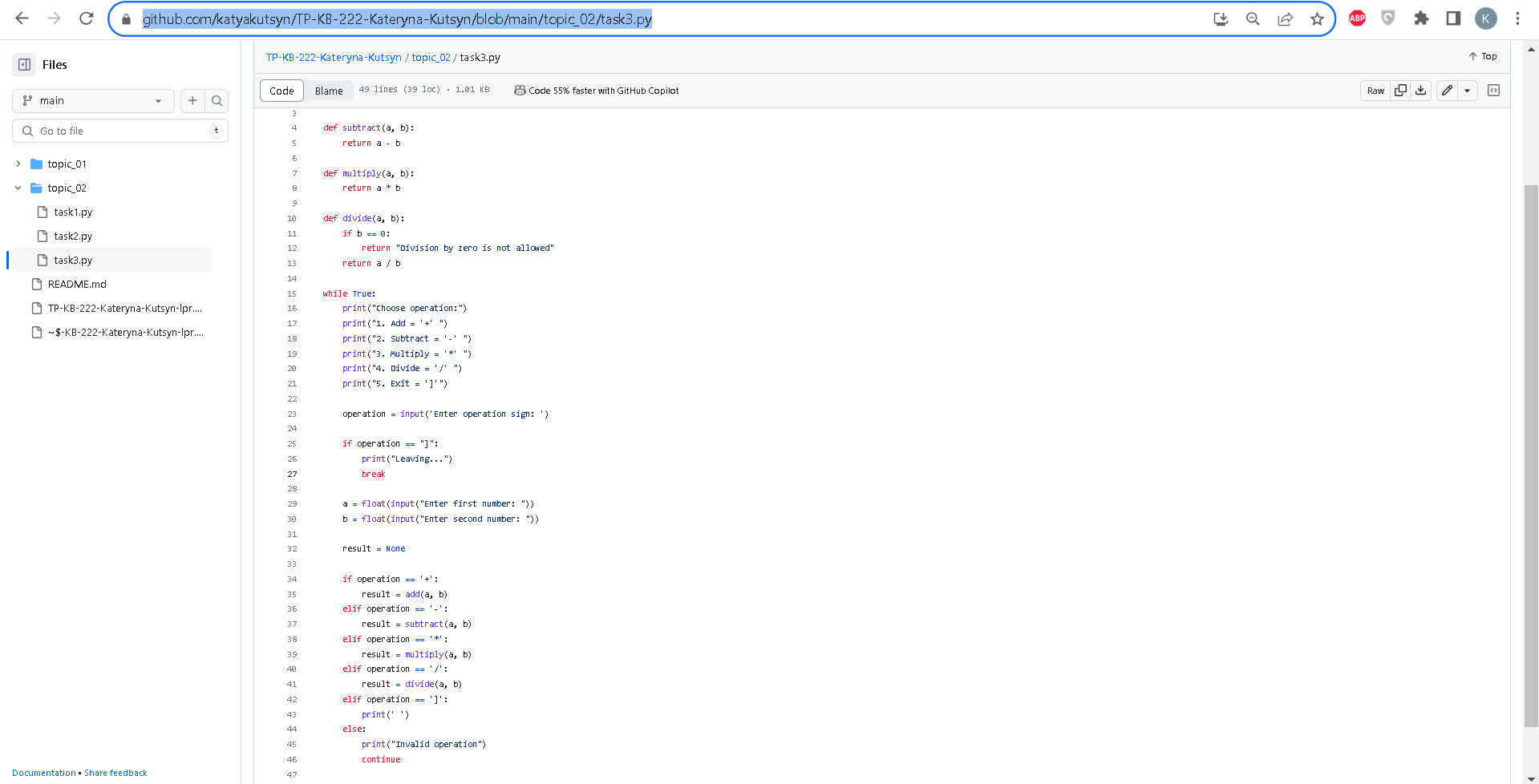
Текст програми:

|  |
| --- |
| def add(a, b):  return a + b  def subtract(a, b):  return a - b  def multiply(a, b):  return a \* b  def divide(a, b):  if b == 0:  return "Division by zero is not allowed"  return a / b  while True:  print("Choose operation:")  print("1. Add = '+' ")  print("2. Subtract = '-' ")  print("3. Multiply = '\*' ")  print("4. Divide = '/' ")  print("5. Exit = ']'")    operation = input('Enter operation sign: ')    if operation == "]":  print("Leaving...")  break    a = float(input("Enter first number: "))  b = float(input("Enter second number: "))  result = None  if operation == '+':  result = add(a, b)  elif operation == '-':  result = subtract(a, b)  elif operation == '\*':  result = multiply(a, b)  elif operation == '/':  result = divide(a, b)  elif operation == ']':  print(' ')  else:  print("Invalid operation")  continue  if result is not None:  print("Result:", result) |

Посилання на github:

<https://github.com/katyakutsyn/TP-KB-222-Kateryna-Kutsyn/blob/main/topic_02/task3.py>

Знімок екрану з посилання на github:



Звіт до Теми №3

Цикли

Під час виконання практичного завдання до Теми №3 було надано варіанти рішення до наступних задач:

**Програма калькулятор**

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.

Хід виконання завдання:

1. Відкрила ВСкод
2. Прочитала завдання
3. Виконала завдання
4. Затестила
5. Закинула в звіт

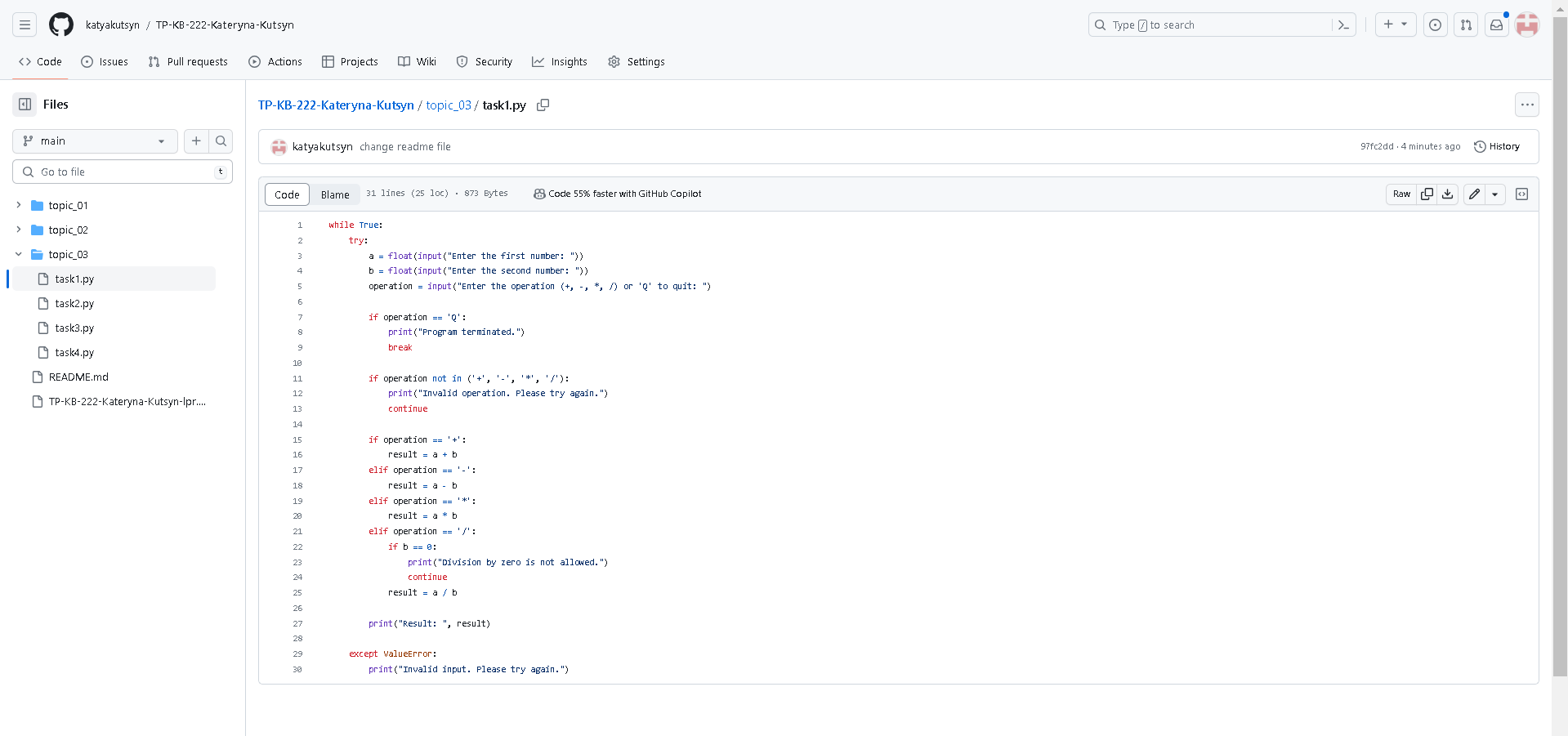
Текст програми:

|  |
| --- |
| while True:  try:  a = float(input("Enter the first number: "))  b = float(input("Enter the second number: "))  operation = input("Enter the operation (+, -, \*, /) or 'Q' to quit: ")  if operation == 'Q':  print("Program terminated.")  break  if operation not in ('+', '-', '\*', '/'):  print("Invalid operation. Please try again.")  continue  if operation == '+':  result = a + b  elif operation == '-':  result = a - b  elif operation == '\*':  result = a \* b  elif operation == '/':  if b == 0:  print("Division by zero is not allowed.")  continue  result = a / b  print("Result: ", result)  except ValueError:  print("Invalid input. Please try again.") |

Посилання на github:

<https://github.com/katyakutsyn/TP-KB-222-Kateryna-Kutsyn/blob/main/topic_03/task1.py>

Знімок екрану з посилання на github:



**Тестування функцій списків**

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

Хід виконання завдання:

1. Відкрила ВСкод
2. Прочитала завдання
3. Виконала завдання
4. Затестила
5. Закинула в звіт

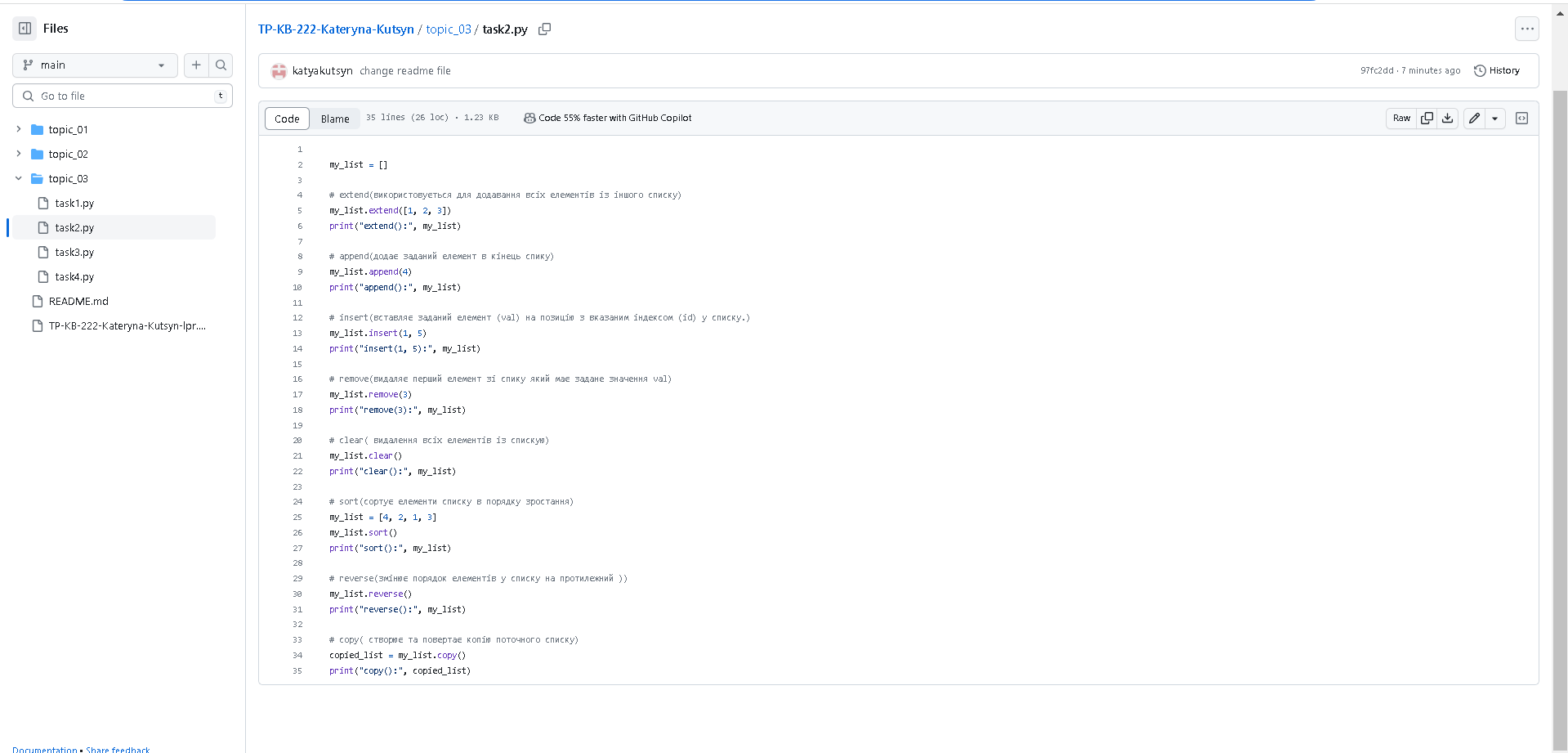
Текст програми:

|  |
| --- |
| my\_list = []  # extend(використовується для додавання всіх елементів із іншого списку)  my\_list.extend([1, 2, 3])  print("extend():", my\_list)  # append(додає заданий елемент в кінець спику)  my\_list.append(4)  print("append():", my\_list)  # insert(вставляє заданий елемент (val) на позицію з вказаним індексом (id) у списку.)  my\_list.insert(1, 5)  print("insert(1, 5):", my\_list)  # remove(видаляє перший елемент зі спику який має задане значення val)  my\_list.remove(3)  print("remove(3):", my\_list)  # clear( видалення всіх елементів із спискую)  my\_list.clear()  print("clear():", my\_list)  # sort(сортує елементи списку в порядку зростання)  my\_list = [4, 2, 1, 3]  my\_list.sort()  print("sort():", my\_list)  # reverse(змінює порядок елементів у списку на протилежний ))  my\_list.reverse()  print("reverse():", my\_list)  # copy( створює та повертає копію поточного списку)  copied\_list = my\_list.copy()  print("copy():", copied\_list) |

Посилання на github:

<https://github.com/katyakutsyn/TP-KB-222-Kateryna-Kutsyn/blob/main/topic_03/task2.py>

Знімок екрану з посилання на github:



**Тестування функцій словників**

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

Хід виконання завдання:

1. Відкрила ВСкод
2. Прочитала завдання
3. Виконала завдання
4. Затестила
5. Закинула в звіт

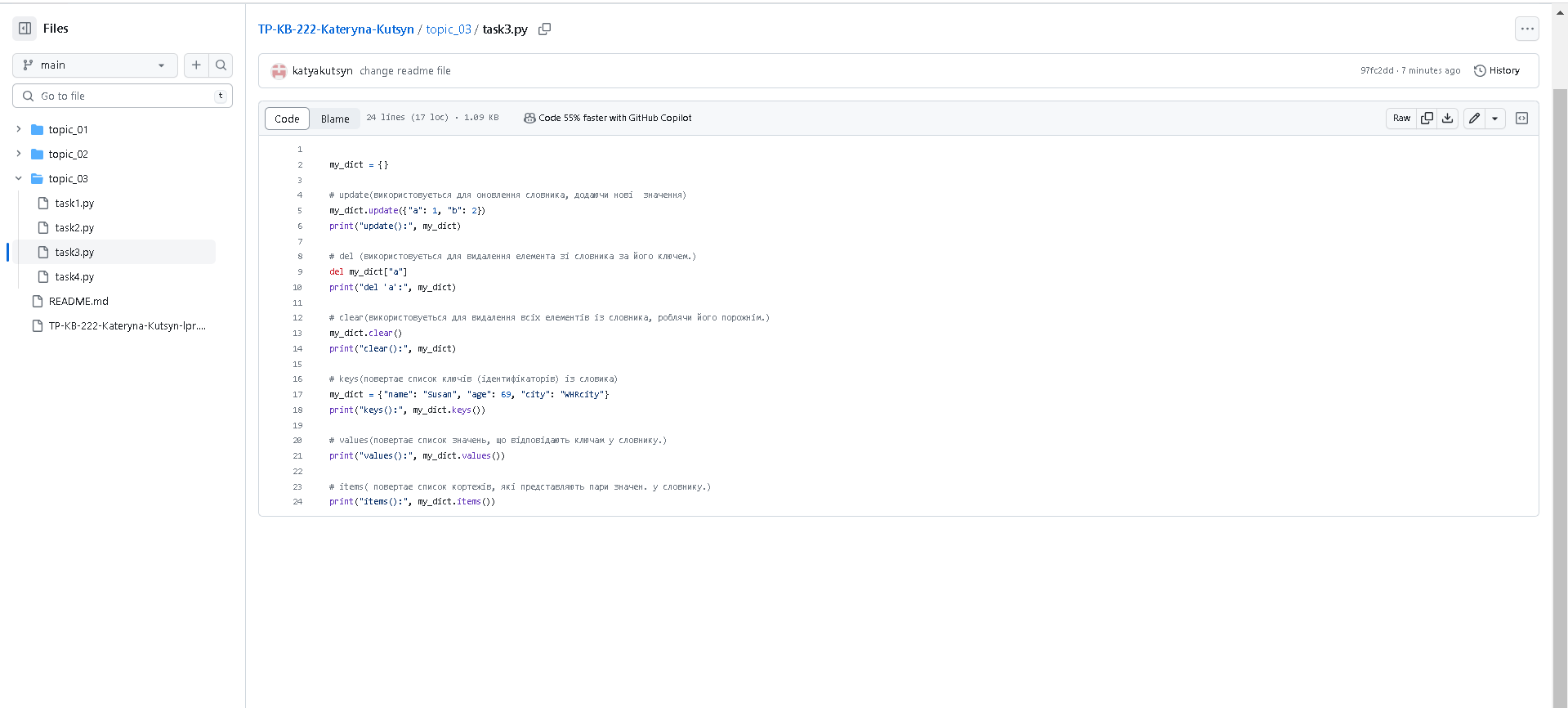
Текст програми:

|  |
| --- |
| my\_dict = {}  # update(використовується для оновлення словника, додаючи нові значення)  my\_dict.update({"a": 1, "b": 2})  print("update():", my\_dict)  # del (використовується для видалення елемента зі словника за його ключем.)  del my\_dict["a"]  print("del 'a':", my\_dict)  # clear(використовується для видалення всіх елементів із словника, роблячи його порожнім.)  my\_dict.clear()  print("clear():", my\_dict)  # keys(повертає список ключів (ідентифікаторів) із словика)  my\_dict = {"name": "Susan", "age": 69, "city": "WHRcity"}  print("keys():", my\_dict.keys())  # values(повертає список значень, що відповідають ключам у словнику.)  print("values():", my\_dict.values())  # items( повертає список кортежів, які представляють пари значен. у словнику.)  print("items():", my\_dict.items()) |

Посилання на github:

<https://github.com/katyakutsyn/TP-KB-222-Kateryna-Kutsyn/blob/main/topic_03/task3.py>

Знімок екрану з посилання на github:



**Функцію пошуку позиції**

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

Хід виконання завдання:

1. Відкрила ВСкод
2. Прочитала завдання
3. Виконала завдання
4. Затестила
5. Закинула в звіт

Текст програми:

|  |
| --- |
| my\_list = ["w", "h", "r"]  while True:  new\_value = input("New value (or 'Q' to exit): ")  if new\_value == 'Q':  break  insert\_index = 0  for elem in my\_list:  if new\_value > elem:  insert\_index += 1  my\_list.insert(insert\_index, new\_value)  print("Result:", my\_list) |

Посилання на github:

<https://github.com/katyakutsyn/TP-KB-222-Kateryna-Kutsyn/blob/main/topic_03/task4.py>

Знімок екрану з посилання на github:

