Міністерство освіти і науки України

Національний університет «Львівська політехніка»

лабораторної роботи №2

з дисципліни «Спеціалізовані мови програмування»

на тему «Основи побудови об’єктно-орієнтованих додатків на Python»

Виконав:

Коханець П. Р.

Перевірив:

Щербак С.С.

Львів 2024

Мета: Розробка консольного калькулятора в об’єктно орієнтованому стилі з використанням класів

Git: <https://github.com/Pivinter/-.git>

Завдання 1: Створення класу Calculator

Створіть клас Calculator, який буде служити основою для додатка калькулятора.



Мал.1.Клас Calculator.

Завдання 2: Ініціалізація калькулятора

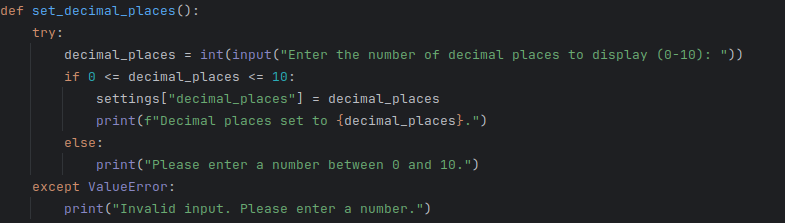
Реалізуйте метод \_\_init\_\_ у класі Calculator для ініціалізації необхідних атрибутів або змінних.



Мал.2.метод \_\_init\_\_ .

Завдання 3: Введення користувача

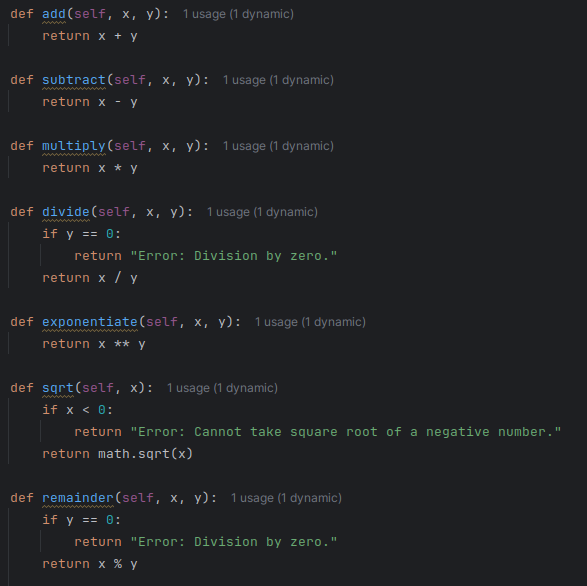
Перемістіть функціональність введення користувача в метод у межах класу Calculator. Метод повинен приймати введення для двох чисел і оператора.



Мал.3. Функціональність.

Завдання 4: Перевірка оператора

Реалізуйте метод у класі Calculator, щоб перевірити, чи введений оператор є дійсним (тобто одним із +, -, \*, /). Відобразіть повідомлення про помилку, якщо він не є дійсним.



Мал.4. Перевірка оператора.

Завдання 5: Обчислення

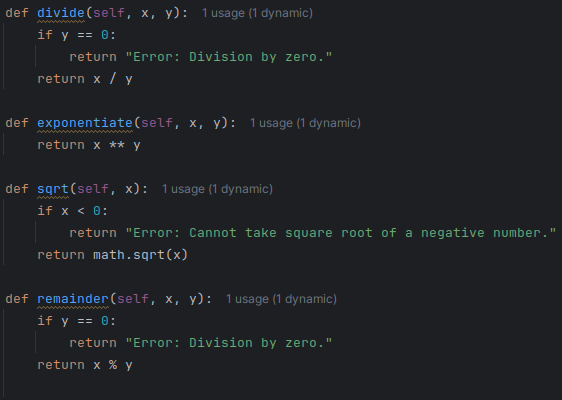
Створіть метод у класі Calculator, який виконує обчислення на основі введення користувача (наприклад, додавання, віднімання, множення, ділення).



Мал.5. бчислення на основі введення користувача.

Завдання 6: Обробка помилок

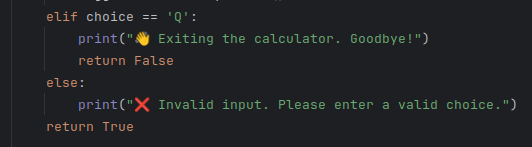
Реалізуйте обробку помилок у межах класу Calculator для обробки ділення на нуль або інших потенційних помилок. Відобразіть відповідні повідомлення про помилку.



Мал.6. Обробка помилок.

Завдання 7: Повторення обчислень

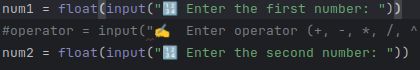
Додайте метод до класу Calculator, щоб запитати користувача, чи він хоче виконати ще одне обчислення. Якщо так, дозвольте йому ввести нові числа і оператор. Якщо ні, вийдіть з програми.



Мал.7. Повторення обчислень.

Завдання 8: Десяткові числа

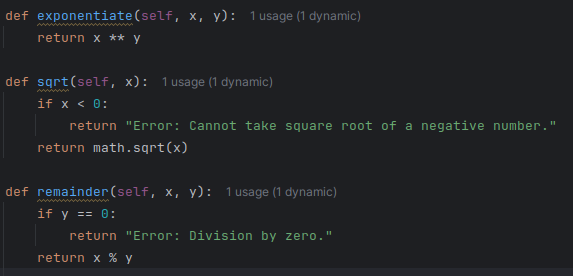
Модифікуйте клас Calculator для обробки десяткових чисел (плаваюча кома) для більш точних обчислень.



Мал.8. Десяткові числа.

Завдання 9: Додаткові операції

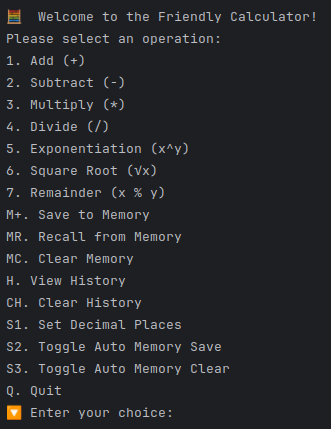
Розширте клас Calculator, щоб підтримувати додаткові операції, такі як піднесення до степеня (^), квадратний корінь (√) та залишок від ділення (%).



Мал.9. Десяткові числа.

Завдання 10: Інтерфейс, зрозумілий для користувача

Покращте інтерфейс користувача у межах класу Calculator, надавши чіткі запити, повідомлення та форматування виводу для зручності читання.



Мал.10.Інтерфейс, зрозумілий для користувача.

Висновок: Виконавши ці завдання, я перетворив консольний калькулятор у об'єктно-орієнтований калькулятор, використовуючи класи в Python. Цей проект допоможе вам вивчити концепції об'єктно-орієнтованого програмування та організацію, зберігаючи функціональність і інтерфейс користувача калькулятора.