Course: Introduction to Programming Using Python

Module 9. Object-Oriented Programming.

Topic. Operator Overloading. Part 5

Task 1

Create a class Number (or use the one you created earlier). The Number class stores one value. Use operator overloading to implement arithmetic operations for working with this number (the operators: +, -, *, /).

Task 2

Create a class Fraction (or use the one you created earlier). Use operator overloading to implement arithmetic operations for working with fractions (the operators: +, -, *, /).

Task 3

Create a class Library. The class is designed to store information about a library (name, address, number of books, etc.) Implement the methods required for this class. Use operator overloading to implement the following arithmetic operations for it:

- + adds the specified value to the number of books;
- - deducts the specified value from the number of books;
- += adds the specified value to the number of books;
- -= deducts the specified value from the number of books.

Use operator overloading to implement the following (comparison of the number of books):

- <;
- >;
- <=;
- >=;
- ==;
- !=.

Task 4

Create a class Date that stores a day, month, year. Use operator overloading to find the difference between two dates (the result should be the number of days between these dates), as well as the operation of increasing the date by a certain number of days.