

Чумин Владимир Владимирович

КЭ-142

Лабораторная № 3

Задание:

Разработка класса Калькулятор для базовых арифметических операций

Описание: Создайте класс Calculator, который может выполнять базовые арифметические операции (+, -, \*, /) с двумя операндами.

Переменные-члены: не требуются. Функции-члены: Методы для каждой операции (+, -, \*, /)

Код :

```
#include <iostream>

class Calculator {
public:
    static int add(int a, int b) {
        return a + b;
    }

    static double add(double a, double b) {
        return a + b;
    }

    static int subtract(int a, int b) {
        return a - b;
    }

    static double subtract(double a, double b) {
        return a - b;
    }

    static int multiply(int a, int b) {
        return a * b;
    }

    static double multiply(double a, double b) {
```

```

        return a * b;
    }
    static double divide(int a, int b) {
        if (b == 0) {
            throw "Error: devision by zero";
        }
        else {
            return a / b;
        }
    }
}

static double divide(double a, double b) {
    if (b == 0) {
        throw "Error: devision by zero";
    }
    else {
        return a / b;
    }
}
};

int main() {
    using namespace std;

    int frstNum, scndNum;
    char operation;

    cout << "Enter first number: ";
    cin >> frstNum;
    cout << "Enter second number: ";
    cin >> scndNum;
    cout << "Enter the operator: ";
    cin >> operation;

    try {

```

```
switch (operation)
{
    case '+':
        cout << frstNum << " + " << scndNum << " =
" << Calculator::add(frstNum, scndNum) << endl;
        break;

    case '-':
        cout << frstNum << " - " << scndNum << " =
" << Calculator::subtract(frstNum, scndNum) <<
endl;
        break;

    case '*':
        cout << frstNum << " * " << scndNum << " =
" << Calculator::multiply(frstNum, scndNum) <<
endl;
        break;

    case '/':
        cout << frstNum << " / " << scndNum << " =
" << Calculator::divide(frstNum, scndNum) << endl;
        break;

    default:
        cout << "Error: unexpected operator";
    }
} catch (const char* msg) {
    cerr << msg << endl;
}

return 0;
}
```

## Результат:

```
vol@vol-All-Series:~/CLionProjects/C++_Labs$ g++ Lab3/lab_3.cpp && ./a.out
Enter first number: 2
Enter second number: 3
Enter the operator: -
2 - 3 = -1
```