

Обобщающий практический урок



ПРЕПОДАВАТЕЛЬ



Артем Гордийчук

Full-stack software engineer

- Более 8 лет опыта работы
- Java, Spring, Hibernate, AWS, Oracle, PostgreSQL
- Проекты связанные с банковской, финансовой деятельностью, e-commerce

artemsgor@gmail.com

www.linkedin.com/in/artem-g-48071a61



ВАЖНО:

- Камера должна быть включена на протяжении всего занятия.
- Если у Вас возник вопрос в процессе занятия, пожалуйста, поднимите руку и дождитесь, пока преподаватель закончит мысль и спросит Вас, также можно задать вопрос в чате или когда преподаватель скажет, что начался блок вопросов.
- Организационные вопросы по обучению решаются с кураторами, а не на тематических занятиях.
- Вести себя уважительно и этично по отношению к остальным участникам занятия.
- Во время занятия будут интерактивные задания, будьте готовы включить камеру или демонстрацию экрана по просьбе преподавателя.

ПЛАН ЗАНЯТИЯ

1. Практическая работа
2. Вопросы



TEL-RAN
by Starta Institute

1

ПРАКТИЧЕСКАЯ РАБОТА

Практическое задание 1

Продолжаем работу над калькулятором.

- Обновите "Калькулятор" до версии 2,
- Обновите функционал до производящего вычисления (+, -, *, /) с произвольным количеством чисел с дробной частью.
- Количество аргументов пользователь вводит с клавиатуры. Оно должно быть от 2 и более.
- Тип данных вводимых аргументов должен быть Double.
- Далее пользователь последовательно вводит значение каждого из аргументов, а затем - знак математической операции, которую необходимо выполнить со значениями введенных аргументов.
- Результат работы программы необходимо вывести в виде вычисляемого выражения и полученного результата с точностью до двух чисел после запятой.

Пример работы программы

Введите количество аргументов (2 и более): 5

Введите значение 0 аргумента: 100

Введите значение 1 аргумента: 2

Введите значение 2 аргумента: 2

Введите значение 3 аргумента: 2

Введите значение 4 аргумента: 2

Введите знак математической операции (+, -, *, /): /

$100,00/2,00/2,00/2,00/2,00=6,25$



Реализация задания 1

```
Scanner scanner = new Scanner(System.in);
scanner.useLocale(Locale.ENGLISH);

System.out.print("Введите количество аргументов (2 и более): ");
int argumentsNumber = scanner.nextInt();
System.out.println();
if(argumentsNumber < 2) {
    System.out.println("Введено не верное кол-во аргументов. Кол-во аргументов должно быть 2 и более");
    System.exit( status: 0); // Завершение работы программы
}
```


Практическое задание 2

Доработайте программу "Калькулятор" (версии 2) таким образом, чтобы она позволяла:

- Вводить знак математической операции перед вводом каждого следующего аргумента.
 - Например, пользователь указывает, что хочет выполнить операцию над тремя аргументами. Программа просит последовательно ввести первый аргумент, знак математической операции, который будет выполнен над значениями первого и второго аргумента, значение второго аргумента, знак математической операции, который будет выполнен над результатом операции между первым и вторым аргументом и третьим аргументом.
 - В результате, программа должна позволять вычислить следующее выражение: $3 * 4 - 10.5 + 2 = 3.5$
- Выводить на экран историю последних 5 вычислений.

Выполнить самостоятельно.

Время выполнения: 35 минут.

Пример работы программы

Введите количество аргументов (2 и более): 2

Введите значение 1 аргумента: 2

Введите знак математической операции (+, -, *, /): +

Введите значение 2 аргумента: 2

2,00+2,00=4,00

История выполненных вычислений:

[2,00+2,00=4,00, null, null]

Выполнить еще одно вычисление? (Y/N):

y



Пример работы программы

Введите количество аргументов (2 и более): 3

Введите значение 1 аргумента: 3

Введите знак математической операции (+, -, *, /): *

Введите значение 2 аргумента: 3

Введите знак математической операции (+, -, *, /): -

Введите значение 3 аргумента: 9

$3,00 * 3,00 - 9,00 = 0,00$

История выполненных вычислений:

[$3,00 * 3,00 - 9,00 = 0,00$, $2,00 + 2,00 = 4,00$, null]



Пример работы программы

Выполнить еще одно вычисление? (Y/N):

y

Введите количество аргументов (2 и более): 3

Введите значение 1 аргумента: 7

Введите знак математической операции (+, -, *, /): *

Введите значение 2 аргумента: 7

Введите знак математической операции (+, -, *, /): -

Введите значение 3 аргумента: 40

$7,00 * 7,00 - 40,00 = 9,00$



Пример работы программы

История выполненных вычислений:

[7,00*7,00-40,00=9,00, 3,00*3,00-9,00=0,00, 2,00+2,00=4,00]

Выполнить еще одно вычисление? (Y/N):

y

Введите количество аргументов (2 и более): 4

Введите значение 1 аргумента: 4

Введите знак математической операции (+, -, *, /): /

Введите значение 2 аргумента: 2

Введите знак математической операции (+, -, *, /): *



Пример работы программы

Введите значение 2 аргумента: 2

Введите знак математической операции (+, -, *, /): *

Введите значение 3 аргумента: 3

Введите знак математической операции (+, -, *, /): -

Введите значение 4 аргумента: 1

$4,00/2,00*3,00-1,00=5,00$

История выполненных вычислений:

$[4,00/2,00*3,00-1,00=5,00, 7,00*7,00-40,00=9,00, 3,00*3,00-9,00=0,00]$

Выполнить еще одно вычисление? (Y/N):

n



Реализация задания 2

```
double[] arguments = new double[argumentsNumber]; // Массив для хранения аргументов
char[] signs = new char[argumentsNumber - 1]; // Массив для хранения знаков математических операций
for (int i = 0; i < arguments.length; i++) {
    System.out.print(String.format("Введите значение %d аргумента: ", i + 1));
    arguments[i] = scanner.nextDouble();
    System.out.println();

    if (i < arguments.length - 1) { // Ввод знаков математической операции для аргументов с первого до предпоследнего
        System.out.print("Введите знак математической операции (+, -, *, /): ");
        signs[i] = scanner.next().charAt(0); // Сохранение введенного знака в массив знаков операций
        System.out.println();
    }
}
```

Практическое задание 3

Доработайте программу "Калькулятор" (версии 2) таким образом, чтобы она позволяла пользователю вводить вычисляемое математическое выражение в виде одной строки. Например, так $10.5 * 5 + 1 - 7.1$ и последовательно выполняла бы указанные арифметические операции над аргументами.

- выполнить рефакторинг программного кода "Калькулятора" таким образом, чтобы каждый логический шаг алгоритма был бы реализован в отдельном методе.
 - создать отдельные методы для выполнения следующих шагов:
 - 1) `String inputExpression()` - для ввода строки (вычисляемого математического выражения) с клавиатуры
 - 2) `double[] parseArguments(String expression)` - для получения из строки математического выражения значений аргументов
 - 3) `char[] parseSigns(String expression)` - для получения из строки математического выражения знаков математических операций
 - 4) `double calculate(double argumentOne, char sign, double argumentTwo)` - для выполнения математической операции над парой аргументов
 - 5) `void printResult(String expression, double result)` - для вывода результата вычисления на экран
- Убедитесь, что приложение работает корректно.
- Соберите дистрибутив приложения (`calculator.jar`), согласно инструкции:
 - https://www.jetbrains.com/help/idea/compiling-applications.html#package_into_jar
- запустите полученный дистрибутив из командной строки своей операционной системы командой:
 - `java -jar calculator.jar`

Пример работы программы

Введите математическое выражение. Например, $10.5*5+1-7.1$:

$2*2$

$2*2=4,00$

История выполненных вычислений:

$[2*2=4,00, \text{null}, \text{null}, \text{null}, \text{null}]$

Выполнить еще одно вычисление? (Y/N):

Y



Пример работы программы

Введите математическое выражение. Например, $10.5*5+1-7.1$:

$11.8*7+2-6.4$

$11.8*7+2-6.4=78,20$

История выполненных вычислений:

[$11.8*7+2-6.4=78,20$, $2*2=4,00$, null, null, null]

Выполнить еще одно вычисление? (Y/N):

N



Реализация задания 3

```
/**
 * Метод выполняет математическую операцию sign над парой аргументов argumentOne и argumentTwo
 */
private static double calculate(double argumentOne, char sign, double argumentTwo) {
    switch (sign) { // Выбор указанной математической операции
        case PLUS_SIGN -> { // Если указан знак + ...
            return argumentOne + argumentTwo; // ... выполняется операция сложения значения очередного аргумента с общим результатом
        }
        case MINUS_SIGN -> { // Если указан знак - ...
            return argumentOne - argumentTwo; // ... выполняется операция вычитания значения очередного аргумента с общим результатом
        }
        case MULTIPLY_SIGN -> { // Если указан знак * ...
            return argumentOne * argumentTwo; // ... выполняется операция умножения значения очередного аргумента с общим результатом
        }
        case DIVIDE_SIGN -> { // Если указан знак / ...
            return argumentOne / argumentTwo; // ... выполняется операция деления значения очередного аргумента с общим результатом
        }
        default -> {
            return 0;
        }
    }
}
```

2

ВОПРОСЫ

Домашнее задание

№1

Подготовить вопросы к лектору для итогового занятия.

№2

Выполнить тест для закрепления материала курса.

Тест прикреплен на платформе, в разделе: Домашнее задание.



ЗАКЛЮЧЕНИЕ

