Методы

№ урока: 8 **Курс:** Java Starter

Средства обучения: Компьютер с установленной IntelliJ IDEA

Обзор, цель и назначение урока

Рассмотрение работы методов.

Изучив материал данного занятия, учащийся сможет:

- Понимать работу методов.
- Понимать работу рекурсии.
- Понимать перегрузку методов.
- Использовать методы с параметрами.

Содержание урока

1. Рассмотрение примера: Перегрузки методов.
2. Рассмотрение примеров: Методы с параметрами.
3. Рассмотрение примеров: Рекурсивный вызов метода.

Резюме

- **Метод** – это именованная часть программы, которая может вызываться из других частей программы столько раз, сколько необходимо.
- **Сигнатура метода** – часть общего объявления метода, позволяющая идентифицировать функцию среди других. В Java, в сигнатуру метода входит Идентификатор метода, тип и количество формальных аргументов.
- **Перегрузка методов** – создание одноименного метода, но с другим типом и/или количеством аргументов.
- Минимизируйте число возвратов из каждого метода. Тяжело понять логику метода, когда при анализе нижних строк приходится помнить о возможных выходах в верхних строках.
- **Рекурсия** – вызов функции (процедуры) из неё же самой, непосредственно (простая рекурсия) или через другие функции (сложная рекурсия). Например, функция A вызывает функцию B, а функция B – функцию A. Количество вложенных вызовов функции или процедуры называется глубиной рекурсии.
- Одна из ключевых задач рекурсивного метода – предотвращение бесконечной рекурсии.
- Реализация рекурсивных вызовов функций, опирается на механизм стека вызовов – адрес возврата и локальные переменные функции записываются в стек, благодаря чему каждый следующий рекурсивный вызов этой функции пользуется своим набором локальных переменных и за счёт этого работает корректно.
- На каждый рекурсивный вызов требуется некоторое количество оперативной памяти, и при чрезмерно большой глубине рекурсии может наступить переполнение стека вызовов. Вследствие этого, обычно рекомендуется избегать рекурсивных программ, которые приводят (или в некоторых условиях могут приводить) к слишком большой глубине рекурсии.
- Не используйте рекурсию для вычисления факториалов и чисел Фибоначчи.
- Так как, методы – это конструкции для выполнения действий, рекомендуется их называть глагольными фразами или глаголами.
- Старайтесь именовать методы в соответствии с задачами, которые они выполняют, а не в соответствии с деталями реализации.
- Для именования методов в Java, рекомендуется использовать соглашение camelCasing. Чтобы выделить слова в идентификаторе, первые буквы каждого слова (кроме первого) сделайте заглавными. Например, `writeLine`, `getType`.
- Язык Java чувствительный к регистру (case sensitivity). Например, `GetType` и `getType` – это разные имена.

CyberBionic systematics

CyberBionic Systematics ® 2016
19 Mariny Raskovoy Str., 5 floor
Kyiv, Ukraine

t. +380 (44) 361-8937
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

- Не используйте символы подчеркивания, дефисы и любые другие неалфавитно-цифровые символы для разделения слов в идентификаторе.
- Описывайте все, что метод выполняет.
- Избегайте невыразительных и неоднозначных глаголов или фраз.
- Для именования метода-функции рекомендуется использовать описание возвращаемого значения. Например: `currentColor()`
- Метод **main** является точкой входа консольного приложения Java. При запуске приложения метод main является первым вызываемым методом.
- В программе Java возможна только одна точка входа. Если в наличие имеется больше одного класса, который имеет метод `main`, то необходимо скомпилировать программу с параметром компилятора /main, чтобы указать, какой метод main нужно использовать в качестве точки входа.
- Метод `main` объявлен внутри класса. Метод `main` должен быть статичным и не иметь атрибута `public`.
- `Main` должен иметь возвращаемый тип `void`. Метод `main` дожен быть объявлен с параметром `String[] args`, который содержит аргументы командной строки. При использовании IntelliJ IDEA для создания приложений, можно добавить параметр вручную или использовать класс настройки IDE для получения аргументов командной строки. Параметры считываются как аргументы нулевого индекса командной строки
- Если значение, возвращаемое методом main, не используется, то указание в качестве возвращаемого типа `void` несколько упрощает код. Однако возврат целого значения позволяет программе передавать информацию о своем состоянии другим программам и скриптам, которые вызывают исполняемый файл.
- Нулевое возвращаемое значение из метода `main` указывает на успешное выполнение программы.

Закрепление материала

- Что такое метод?
- Чем отличаются функции и процедуры?
- Что такое перегрузка метода?
- Что такое рекурсия и какие виды рекурсии вы знаете?
- Какие правила именования применимы к методам?
- Можно ли выполнить перегрузку метода main?

Дополнительное задание

Задание
Используя IntelliJ IDEA, создайте класс **Calculator**.
Создайте метод с именем `calculate`, который принимает в качестве параметров три целочисленных значения и возвращает значение каждого аргумента, деленного на 5.

Самостоятельная деятельность учащегося

Задание 1
Выучите основные конструкции и понятия, рассмотренные на уроке.

Задание 2
Используя IntelliJ IDEA, создайте класс **Bank**.
Представьте, что вы реализуете программу для банка, которая помогает определить, погасил ли клиент кредит или нет. Допустим, ежемесячная сумма платежа должна составлять 100 грн. Клиент должен выполнить 7 платежей, но может платить реже большими суммами. Т.е., может двумя платежами по 300 и 400 грн. Закрыть весь долг.
Создайте метод, который будет в качестве аргумента принимать сумму платежа, введенную экономистом банка. Метод выводит на экран информацию о состоянии кредита (сумма задолженности, сумма переплаты, сообщение об отсутствии долга).

CyberBionic systematics
CyberBionic Systematics ® 2016
19 Mariny Raskovoy Str., 5 floor
Kyiv, Ukraine
t. +380 (44) 361-8937
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

Задание 3
Имеется N клиентов, которым компания производитель должна доставить товар. Сколько существует возможных маршрутов доставки товара, с учетом того, что товар будет доставлять одна машина?
Используя IntelliJ IDEA, создайте класс **Delivery**.
Напишите программу, которая будет рассчитывать и выводить на экран количество возможных вариантов доставки товара. Для решения задачи, используйте факториал N!, рассчитываемый с помощью рекурсии. Объясните, почему не рекомендуется использовать рекурсию для расчета факториала. Укажите слабые места данного подхода.

Рекомендуемые ресурсы

Метод main на примере программы “Hello World!”
https://docs.oracle.com/javase/tutorial/getStarted/application/

Определение, именование и перегрузка методов в JAVA
https://docs.oracle.com/javase/tutorial/java/javaOO/methods.html

CyberBionic systematics
CyberBionic Systematics ® 2016
19 Mariny Raskovoy Str., 5 floor
Kyiv, Ukraine
t. +380 (44) 361-8937
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com