



ОНЛАЙН-ОБРАЗОВАНИЕ

План занятия

- Знакомство
- Орг. вопросы
 - расписание
 - дз
- Gradle (часть первая)





Давайте знакомиться

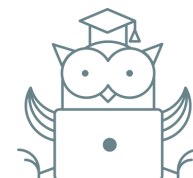
Какой у вас опыт в IT?

Сколько лет с Java?

Вы смотрели ?

День Открытых Дверей курса «Разработчик Java»

<https://youtu.be/0-6OYFAgPOA>



Петрелевич Сергей

Java-разработчик в Raiffeisen Bank, Capital Markets

В отрасли с 2003 года.

Более 10 лет в разработке финансового ПО.

Занимался разработкой "коробочных" систем для внешних заказчиков, разработкой систем для внутренних заказчиков.



Oracle Certified Professional, Java SE 8 Programmer



Александр Оруджев

Инженер-программист. В отрасли с 2006 года.

Входит в состав команды, занимающейся разработкой ИС для информационного сопровождения бурения нефтяных и газовых скважин.



Павел Стрекалов

- АО Гринатом, ведущий специалист
- Профессионально в ИТ с 2006 г
- Окончил Национальный исследовательский университет "МИЭТ", 2006 г
- MS SQL с 2006, .NET с 2009, Java с 2015
- Наставник в OTUS на курсах "Разработчик Java", "Реляционные базы данных", преподаватель на курсе "MS SQL Server разработчик"



Виталий Куценко

- В 2004 году окончил Химический Факультет МГУ им М.В.Ломоносова, в 2005 - вечерний факультет МИФИ по специальности инженер-системотехник
- Участвовал в проектах разных размеров и нагрузки, от небольших на несколько рабочих мест одного предприятия до обработки потоков данных крупнейших банков мира.
- Работал в таких компаниях как Diasoft, 1С, Luxoft, Росгосстрах, Сбербанк.



Вячеслав Лапин

- Учился ВМиК МГУ, Горный
- Работал Luxoft, Lanit, Kaspersky, 1С, AT-Consulting, КРОК, EPAM
- Вошёл в IT в 2003'м...
- Выступал Joker, JPoint, Java-Days...
- Интересуюсь FRP, внедрением инноваций



Орг. вопросы

Расписание занятий

вторник – 20:00

пятница – 20:00

Следующее занятие – 6 июля (вторник).



Орг. вопросы

Будет ли запись вебинаров?

Запись будет, даже если ее не будет 😊

Off-line участники также важны как и on-line.



Орг. вопросы

ДЗ – очень важная часть курса.

ДЗ желательно делать до следующего вебинара, но не обязательно.

Типовой срок проверки: 2-3 дня,

но может быть задержка до недели.

Проверяем внимательно, но без лишнего садизма.

ДЗ отправлять в виде ссылки на Pull Request в github.



ДЗ к этой лекции

- 1) Создайте аккаунт на github.com (если еще нет)
- 2) Создайте репозиторий для домашних работ
- 3) Сделайте checkout репозитория на свой компьютер
- 4) Создайте локальный бранч `hw01-gradle`
- 5) Создать проект gradle
- 6) В проект добавьте последнюю версию зависимости
`<groupId>com.google.guava</groupId>`
`<artifactId>guava</artifactId>`
- 7) Создайте модуль `hw01-gradle`
- 8) В модуле сделайте класс `HelloOtus`
- 9) В этом классе сделайте вызов какого-нибудь метода из `guava`
- 10) Создайте "толстый-jar"
- 11) Убедитесь, что "толстый-jar" запускается.
- 12) Сделайте pull-request в gitHub
- 13) Ссылку на PR отправьте на проверку (личный кабинет, чат с преподавателем).

При желании, можно сделать maven-проект и далее на курсе работать с maven-ом.
Для Maven инструкция аналогичная (просто в тексте замените Gradle на Maven).



Java

Какую версию будем использовать ?

Примеры будут на 15.

Можно использовать 11, если сильно хочется, но зачем?

Java 11 – текущий LTS, поэтому идеологически ориентируемся на нее.



Инструменты

GIT

ДЗ должны быть в gitHub.

У каждого студента свой репозиторий.

В каждом репозитории должен быть файл README.md

За основу возьмите README.md с примерами и допишите:

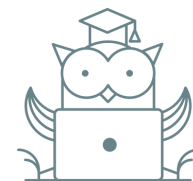
Студент:

Ivan Petrov (Иван петров)

ivan@yandex.ru (опционально, можно не писать)

https://github.com/petrellevich/otus_java_2021_06

ОБРАТИТЕ ВНИМАНИЕ НА файл .gitignore



Инструменты

Idea – стандартная IDE, подойдет любая, включая vi.



Gradle

<https://gradle.org/>

Gradle is an open-source build automation tool focused on flexibility and performance.

Gradle build scripts are written using a Groovy or Kotlin DSL.

Gradle очень быстрый и очень гибкий (!).

Почему не maven?

Для чего нужен?

- управление зависимостями
- сборка проекта





Какими зависимостями управляет?

Нужные проекту jar-файлы можно скачать «руками»,
а можно указать gradle-у, что будет использовано.
Он сам скачает, если надо.



Что собирает gradle?

Gradle собирает «финальный артефакт», например jar-файл.
Jar-файл – это zip-файл.

Jar-ник может быть:

«ТОНКИЙ» и «ТОЛСТЫЙ».

В общем случае файлов может быть много разных.

Посмотрим на примеры таких файлов...



Как установить gradle?

1. Можно скачать с официального сайта и установить как обычную программу.
2. Можно не устанавливать и использовать gradlew



Локальные файлы gradle

Gradle wrapper – «переносимый запускатор»

gradle-wrapper.properties – файл с настройками

settings.gradle – настройки проекта

build.gradle – описывает как проект собирать

Посмотрим как это выглядит в жизни...



Gradle. Как работает

Два ключевых понятия:

- project
- task

Работа Gradle – это выполнение последовательности задач (task).

Перед началом работы задачи выстраиваются в граф.

Граф строится на основе зависимостей, однако задачи могут быть и независимы.

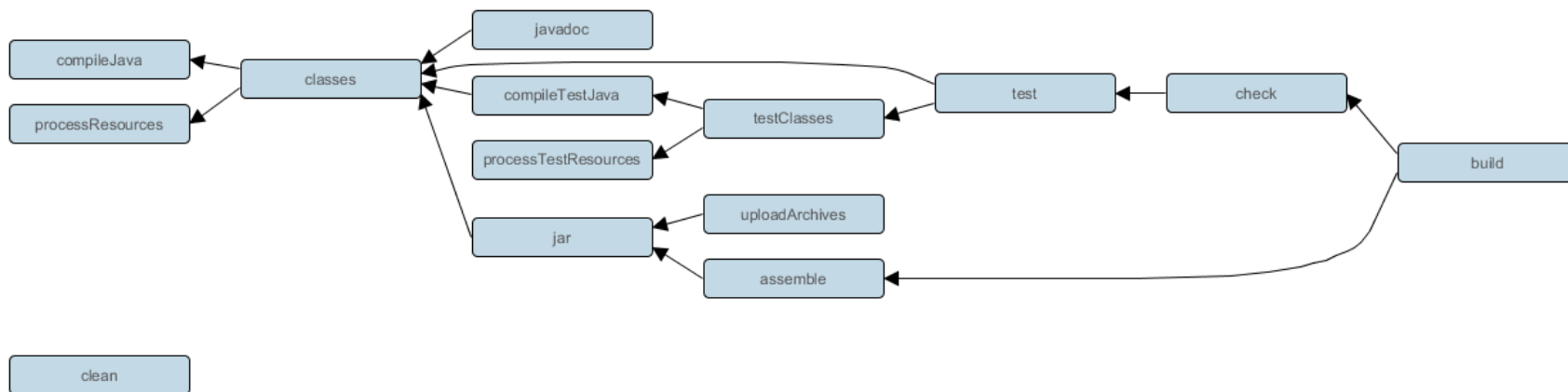
Плагины расширяют список задач, например:

[The Java Plugin](#) (поставляет набор нужных задач)



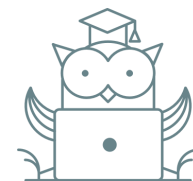
Java Plugin. Задачи

Пример зависимых задач:



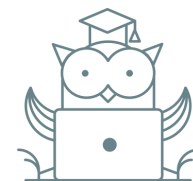
Gradle. Демо

Как создать Gradle-проект



Gradle. Демо

Посмотрим на весь пример в сборе...





Gradle. Горячие команды

`gradle test`

`gradle clean build`





Основной вопрос лекции

Что такое gradle?

Для чего gradle нужен?



Вопросы?



Опрос !!!

Пожалуйста, пройдите опрос.



Спасибо
за внимание!

