

Курс будет осенью 2019г. на матмехе. Также вероятно будет весной на РТФ.

Ориентируемся на студентов 3-4 курса, которые уже умеют как-то прогать на Java, C#, Python в императивном/ООП стиле.

Цели курса

Программа минимум - начать вести курс по скале, порассказывать что-нибудь полезное и интересное, подготовить студентов к найму к нам в департамент, чтобы они были уже более менее знакомы со скалой.

Программа максимум - сделать крутой курс с домашками и грамотной подачей материала. Также в перспективе возможен сценарий вывода в онлайн и автоматизацию проверок домашних/задачек (масштабируемость).

По подготовке

Тайминг - порядка 80 минут.

Хорошо бы в каждой лекции приводить практические интересные примеры на которых будет видно как можно классно применять фишки и концепции которые рассказываются на лекции.

Хорошо бы подготовить набор упражнений/домашних заданий по каждой лекции на практическое применение нового материала.

Перед датой лекции будет обязательный ее прогон. Если надо - больше одного.

Программа курса (драфт 2.0)

1. Вводная лекция

Цель: рассказать про скалу вообще, чем отличается от других языков, подчеркнуть преимущества, познакомить с тулингом для старта чтобы человек мог написать хелло ворлд.

Тезисы:

- Введение.
- Основы языка (литералы, expressions, функции - first class citizen, и тд).
- Тулинг для старта (REPL, worksheets, IDE).

2. Императивное программирование (if, while, for, arrays, tuples и т.д.)

Цель: научить писать тело функции и императивный код, чтобы человек мог написать какой-то простой алгоритм (сортировку, перемножение матриц и т.д.).

Примечание: В целом студенты знакомы со всем этим, т.к. программируют на Java, C#, Python, но думаю будет полезным привыкнуть к синтаксису scala, порешать задачки на нем.

Тезисы:

- if, while, for
- операторы

- базовые типы и преобразования между ними
- простая работа с arrays/tuples/lists

3. Типы

Цель: познакомить с системой типов. Заложить понимание пользы продвинутой системы типов, показать в чем польза кодирования каких-то вещей в типах.

Тезисы:

- особенности системы типов в Scala(статичность, строгость, проверка компилятором)
- Any, AnyRef, AnyVal, Unit, Nothing, Null
- вывод типов - рассказать базово просто как фича языка

4. Практика после блока №1

5. ООП

Цель: научить писать ООП код на скале, показать фичи и возможности.

Тезисы:

- class
- trait, abstract class
- case class
- companion object
- self type (?)

6. Параметрический полиморфизм (Generics, вариантность, и проч.)

Цель: познакомить с концепцией параметрического полиморфизма, научить использовать дженерики, показать их пользу.

Тезисы:

- тайп параметры в функциях (методах)
- рассказать про дженерики (в классах)
- variance
- баунды

7. Collections (Всё про библиотеку коллекций в скале и работу с ними)

8. Практика после блока №2

9. Функции и ФП ч.1 (введение, база, простые конструкции)

Цель:

Тезисы:

- введение в ФП вообще и особенности скалы

- чистота, иммутабельность, referential transparency
- функции в scala подробно (?) (каррирование, apply compose, отличие от методов, тип и тд)
- рекурсия
- паттерн матчинг (применение, unapply/extractors (?))

10. ФП ч.2 (тайпклассы)

Цель:

Тезисы:

- type classes
- ... (наверное что-то еще сюда стоит добавить. хотя можно и всю лекцию спокойно про тайпклассы рассказывать)

11. ФП ч.3 (монады)

Цель:

Тезисы:

- монады теория (весьма упрощенно)
- базовые монады scala (Option, List, Either, ...)
- познакомить с IO/Cats/Scalaz/ZIO (?)

12. Практика после блока №3

Инфа в помощь лекторам

Ресурсы (обзор)

- курс Олега Нижника (Тинькофф) на Stepik
<https://stepik.org/course/16243/syllabus>
 Довольно много, подробно, открыто. Но сжато, не подробно, не глубоко - можно брать от туда материал и дополнять его. Можно брать упражнения/задачи/тесты - но придется это "оцифровывать" в свой формат.
- курс от Luxsoft
https://www.luxoft-training.ru/kurs/yazyk_scala_dlya_java-razrabotchikov.html
 Содержание не знаю, но выложена программа - хорошая структура курса и подробность тем. Можно ориентироваться на последовательность и на перечень подтем и фич про которые стоит рассказывать.
- scala exercices от 47degrees
https://www.scala-exercises.org/scala_tutorial/terms_and_types

https://www.scala-exercises.org/std_lib/asserts

Хорошие примеры и подача материала. Можно брать и адаптировать темы и их подачу. Можно брать задачки.

- scala school от Twitter

https://twitter.github.io/scala_school/

Можно использовать материал для своих лекций, немного но может помочь.

- курс от Головача на udemy

<https://www.udemy.com/scala-for-java-developers-ru/>

Доступа нет, есть список тем. Можно посмотреть.

- Youtube курс от Головача

<https://www.youtube.com/playlist?list=PLoij6udfBnchT7LcACtKD8zWGEpSduNqO>

Можно посмотреть видосы в качестве примера подачи материала.

- TOUR OF SCALA (docs.scala-lang.org)

<https://docs.scala-lang.org/ru/tour/tour-of-scala.html>

Хороший материал, можно оттуда брать под конкретные лекции.

- хорошие книжки/курсы <https://underscore.io/books/>

в частности могут пригодиться некоторые параграфы из

<https://underscore.io/books/essential-scala/>

Задачки по scala

hackerrank

99 scala problems

scala puzzlers

<https://stepik.org/course/16243/syllabus>

<https://www.scala-exercises.org>