Программа курса «Врубиться в Ruby»

# Темы 1–7

## Занятие 1: Базовые навыки и построение процесса разработки –16.05.2016

* Знакомство
* История языка Ruby
* Ruby и Ruby on Rails что это такое и чем они отличаются
* Сравнение Ruby и других языков программирования, плюсы и минусы
* Различные интерпретаторы Ruby mri, jruby, ruby enterprise edition
* Управление версиями Ruby с помощью rbenv и rvm
* Первое приложение на Ruby
* утилита irb
* Система контроля версия git основные принципы работы
* git flow
* семантические коммиты
* Continuous Integration
* Основы Scrum
* Выбор проекта для работы в процессе обучения Домашнее задание

## Занятие 2: Основы Ruby – 19.05.2016

* Базовые типы данных в ruby числа, текст, булевые значения
* Коллекции индексные и ассоциативные массивы (хэши), диапазоны
* идентификаторы (символы)
* комментарии в коде
* регулярные выражения, сервис rubular
* Операторы
* Условные предложения if/else/elsif, а также “условие ? true : false”
* Циклы while/for
* Управление ходом выполнения return/break/next/redo
* Написание приложения по изученному материалу Домашнее задание

## Занятие 3: Объектно-ориентированное программирование (ООП) – 23.05.2016

* Что такое ООП
* Инкапсуляция, наследование, полиморфизм
* Классы
* Модули
* Методы
* public, private и protected области видимости
* Зона видимости переменной
* Замыкания и lambda-функции
* Написание приложения по изученному материалу
* Домашнее задание

## Занятие 4: Чтение и запись данных, Exceptions – 26.05.2016

* Работа с потоками файлы и ввод с клавиатуры, что между ними общего и какие отличия
* Обработка различных типов файлов
* Исключения что это такое и зачем они нужны
* Обработка исключений в ruby-коде
* Порождение исключений методом throw
* Написание приложения по изученному материалу Домашнее задание

## Занятия 5 и 6: Основы web-программирования. Что такое HTML? – 30.05.2016

* Что такое CSS?
* Что такое JavaScript и CoffeeScript, чем они отличаются?
* Как использовать css-фреймворки? twitter bootstrap, zurb foundation, semantic ui и другие
* Протокол http и как он работает. Методы протокола GET, POST, PUT, DELETE
* Будущее http http 2.0
* Что такое REST и как он реализуется в веб-приложениях
* Написание приложения по изученному материалу Домашнее задание

## Занятие 7: Базы данных – 02.06.2016

* Что такое SQL
* Что такое NoSQL
* Что такое KVS
* Как хранятся данные в БД
* Как запрашивать и обновлять данные в БД
* Как работают кластеризация и шардинг
* Какие еще возможности есть у БД
* Чем хорош RethinkDB
* основы ReQL
* Кластеризация в RethinkDB
* Написание приложения по изученному материалу Домашнее задание

# Темы 8–14

## Занятие 8: Основы Ruby on Rails

* История фреймворка, его авторы
* Создание первого Rail-приложения
* Структура rails-приложения
* Что такое MVC и зачем это надо
* Что такое scaffold
* Написание приложения по изученному материалу Домашнее задание

## Занятия 9: Active Record

* Что такое ORM
* Что такое ActiveRecord
* Что такое модели
* Что такое миграции
* CRUD-операции с данными
* Валидация данных стандартные хэлперы
* Валидация данных пишем свой хэлпер
* Валидация по условию
* Обработка ошибок валидации
* Коллбэки что это и зачем это нужно
* Написание приложения по изученному материалу
* Домашнее задание

## Занятия 10 и 11: Advanced Active Record

* Ассоциации one to one, one to many, many to many
* Полиморфные ассоциации
* Self Joins (замыкание на себя)
* Построение ActiveRecord запросов
* Joins
* Scopes
* Eager Loading
* Дополнительные сведения
* Написание приложения по изученному материалу Домашнее задание

## Занятие 12 и 13: Action View

* Action View что это такое и для чего надо
* erb, haml, slim, builder
* partials
* layouts
* какие есть помощники (helpers) и зачем они нужны
* coffee script и javascript
* css, less и sass
* asset pipeline
* Формы в rails-приложениях
* Написание приложения по изученному материалу Домашнее задание

## Занятие 14: Маршрутизация в rails

* Что такое CRUD и как это соотносится с REST
* Как пишутся routes
* namespace и scope
* Вложенные маршруты Shallow nesting
* routing concerns
* Расширение REST Поддомены
* Дополнительные сведения
* Написание приложения по изученному материалу Домашнее задание

# Темы 15–20

## Занятие 15: Action Controller

* Вспоминаем rails routes actions
* передаваемые параметры сессии
* flash зачем это нужно
* cookies
* обработка ошибок, кастомные шаблоны 403, 404 и 500
* Написание приложения по изученному материалу Домашнее задание

## Занятие 16: i18N

* Как реализована поддержка мультиязычных приложений в rails
* Готовим приложение к локализации
* Определение языка пользователя
* Поддержка многоязыковых моделей с помощью gem’ов globalize и multilang-hstore
* Написание приложения по изученному материалу Домашнее задание

## Занятие 17: Action Mailer

* Отправка писем в rails
* Получение писем в rails
* отправка писем с помощью mandrill
* создание рассылок с помощью mailchimp
* Написание приложения по изученному материалу Домашнее задание

## Занятие 18: Active Job, Delayed Job, Sidekiq и другие инструменты для асинхронной работы

* Чем отличается синхронный режим работы от асинхронного
* Что такое Active Job и какие у него есть коннекторы
* Основы Delayed Job
* Работаем с Sidekiq
* гем Whenever
* Написание приложения по изученному материалу Домашнее задание

## Занятие 19: Тестирование приложений (основы)

* Зачем тестировать приложения
* Какие есть инструменты для тестирования
* Что такое TDD
* Что такое BDD Rspec
* Cucumber и Turnip
* Написание приложения по изученному материалу Домашнее задание

## Занятие 20: Безопасность web-приложений

* Какие основные типы атак на web-приложения самые актуальные
* Атаки на сессию Session Hijacking, Session Fixation
* Cross-Site Request Forgery (CSRF)
* Injection-атаки
* Brute Forcing
* Что такое Captcha и какие виды капчи существуют
* Дополнительные сведения
* Написание приложения по изученному материалу Домашнее задание

# Темы 21–27

## Занятие 21 и 22: Аутентификация с помощью Devise, Управление доступом cancancan и pundit. Установка devise в проект

* Кастомизация devise
* аутентификация через devise + omniauth
* Написание приложения по изученному материалу Домашнее задание

## Занятие 23: Основы AJAX в rails-приложениях

* Query pjax
* гем gon
* JSON и msgpack
* Написание приложения по изученному материалу Домашнее задание

## Занятие 24: Поиск

* Используем ransack
* Поиск с помощью elasticsearch
* Написание приложения по изученному материалу Домашнее задание

## Занятие 25: API

* Что такое API и зачем это нужно
* Написание API с помощью Action Controller
* использование grape для написания API
* Написание приложения по изученному материалу Домашнее задание

## Занятие 26: Deploy, Docker

* Как организовывать окружения test/staging/production
* Готовим первый тестовый сервер
* Пишем первый deploy-скрипт для capistrano
* Что такое Docker
* чем различаются images и контейнеры
* что такое docker-compose
* Как использовать Docker для Ruby on Rails
* Что такое Docker Hub
* Написание приложения по изученному материалу Домашнее задание

## Занятие 27: Автоматизация настройки серверов с помощью Ansible

* Что такое Ansible
* Учимся писать playbooks
* Создаем роли для зависимостей
* Ansible galaxy
* что такое кэширование и зачем оно нужно
* Какие виды кэширования есть
* использование ruby ffi для оптимизации низкопроизводительных кусков кода
* Tarantool для rails-приложений
* Elixir и Phoenix Framework как дальнейшее развитие MVC-фрэймворков
* Написание приложения по изученному материалу
* Домашнее задание