

Разработчик Python

Best Practice по решению прикладных задач и освоению инструментов, применяемых программистом при разработке инфраструктурных решений, веб-приложений, систем контроля качества и аналитических систем

Продолжительность

5 месяцев, 4 часа в неделю

Начало занятий

20 марта

1 Advanced

1 **Advanced basics. Часть 1**

Кодировки, Unicode в Python 2, coercion. Floating point numbers, IEEE-754, особенности реализации, основные ошибки и особенности использования. Itertools, iterable, iterator, iterator protocol, sequence protocol. Generators: pipelines, routing, coroutines.

Домашние задания

1 ДЗ-1: Log Analyzer

(1) Несколько упражнений на тему занятия: итераторы, генераторы, декораторы; (2) Делаем анализатор логов веб-сервера.

2 **Advanced basics. Часть 2** Функциональное программирование, рекурсия и ее ограничения, tail recursion optimization. Scoping, namespaces, замыкания и их устройство. Декораторы функции и классы, декораторы с аргументами, вложенные декораторы.

3 **Internals. Часть 1** CPython, внутренности и архитектура, байткод, stack virtual machine, интерпретация. Имплементация основных типов данных: строки, числа, списки, словари. Оптимизации.

Домашние задания

1 ДЗ-2: CPython (опционально)

Добавляем новый опкод и пару операторов, собираем свой Python. Понадобятся знания C.

4 **Internals. Часть 2** Memory model, allocation, reference counting, garbage collection. GIL, реализация и последствия, методы обхода.

5 **ООП. Часть 1** Object model, new-style classes, MRO, super. Множественное наследование, slots, new, name mangling. Декрипторы, data и non-data дескрипторы, descriptor protocol, property.

Домашние задания

1 ДЗ-3.0: Scoring API

Создаем декларативный язык описания и систему валидации запросов к HTTP API

6	ООР. Часть 2	Интроспекция. Магические методы. Модуль ABC, абстрактные классы. Metaclasses и "черная магия". Design patterns, ключевые принципы, примеры.
7	Testing. Часть 1	<p>Пирамида тестирования. Test design, coverage. Unit testing, integration testing. Mocking, monkey patching. Фикстуры. Нагрузочное тестирование.</p> <p>Домашние задания</p> <p>1 ДЗ-3.1: API Testing</p> <p>Дописываем тесты API</p>
8	Testing. Часть 2	<p>Автоматизация тестирования. Continuous integration, continuous delivery, continuous deployment. Canary deployment, feature flags, staged rollouts.</p> <p>Документирование.</p>
9	Automatization. Часть 1	<p>Базы данных, коннекторы, ping-reconnect, конкурентное использование, connection pooling, "отстреливание" долгих запросов. Кэширование. Networking, socket, UDP, TCP/IP. HTTP, timeouts, requests.</p> <p>Домашние задания</p> <p>1 ДЗ-4: Web Server</p> <p>Создаем свой сервер на "ванильном" Python, частично реализующий протокол HTTP (будет корректно отдавать страницу wikipedia).</p> <p>Проводим нагрузочное тестирование.</p>

10 **Automatization.**
Часть 2

Архитектуры web-server'ов. Демонизация, конфиги, логирование. Setup tools и дистрибуция, менеджмент зависимостей. Пакеты и пакетные менеджеры, сборка.

- 1 Dynamic Web**

Dynamic Web, CGI, FastCGI, WSGI, Gunicorn. uWSGI, обзор основных особенностей. Tiered architecture.

Домашние задания

 - 1 ДЗ-5: uWSGI Daemon/ Django Tutorial (опционально)

Пишем стандартного "промышленного" демона, который будет отвечать по HTTP, ходить в базу, писать логи и собираться в пакет. Те, кто не знаком с Django, проходят tutorial.

- 2 Django**

Паттерн MVC\MTV, coding style, настройка окружения, структура Django проекта, конфигурация проектов, зависимости, маршрутизация URL.

- 3 ORM и "зло"**

Модели, CRUD, Query sets, lazy evaluation, prefetch\preload, join'ы, сложные запросы и raw SQL. Object managers.

Домашние задания

 - 1 ДЗ-6.0: Django project

Создание web-приложения аналога Stack Overflow. Определяемся со структурой и схемой данных.

- 4 Database**

Индексы. Транзакции и уровни изоляции. Миграции схемы и миграции данных, большие объемы данных. Репликация, перенос сложных запросов на slave'ы, распределение нагрузки. Шардирование. NoSQL.

5	Views	Views, представления, виды представлений, function/class based views, generic views. Домашние задания 1 ДЗ-6.1: Django project Начинаем рисовать красивые странички
6	Формы	Формы, поля, процесс валидации, model form. Widgets. Templates, язык шаблонов, архитектура шаблонизатора. Фильтры, тэги. Jinja2
7	REST API. Часть 1	Принципы REST, архитектурные стили, ограничения. Проектирование API, ресурсы, методы ошибки, версионирование. Django REST framework. Домашние задания 1 ДЗ-6.2: Django REST API (опционально) Добавляем к разрабатываемому приложению API.
8	REST API. Часть 2	Rate limiting. Документация, swagger. Web Performance. Фронтэнд оптимизация, масштабирование бекэнда.

3 Data engineering

- 1 **NumPy. Часть 1** IPython, базовое использование, интроспекция, поиск, история, макросы, магические методы, взаимодействие с ОС, разработка. NumPy. ndarray, индексирование, маски, векторизация. Universal functions.

Домашние задания

- 1 ДЗ-7: LogRegression

Дописываем логистический регрессор, используем его для классификации отзывов о еде из Amazon.

- 2 **NumPy. Часть 2** Reshaping, broadcasting, structured и record массивы, хранение и загрузка данных. Внутренности ndarray. Мемmap, HDF5.
-

- 3 **Pandas** Pandas. Series. Dataframe, иерархические индексы, missing data, агрегация.

Домашние задания

- 1 ДЗ-8: Open Data Analysis (опционально)

Выбираем один из открытых dataset'ов и анализируем его с помощью pandas в IPython notebook.

- 4 **Matplotlib** Построение графиков, гистограммы, subplots, аннотации, стили. Data visualization, принципы правильных визуализаций.

4 Highload

1 Concurrency. Часть 1

Concurrency, parallelism. IO/CPU bound задачи.
Multithreading, sharing, communication.

Домашние задания

1 ДЗ-9: MemcLoad

Реализуем конкурентную заливку данных в memcache'ы

2 Concurrency. Часть 2

Multiprocessing, IPC, shared memory, Manager.
Distributed computing

3 C extensions

Написание расширений на C, C API.

Домашние задания

1 ДЗ-10: Protobuf (de)serializer (опционально)

Пишем свое расширение, которое будет писать файлы с protobuf сериализованным содержимым. Понадобятся знания C.

4 ffi. Cython. PyPy

5 Async. Часть 1

Generators, coroutines, yield from. Event loop. Future.

Домашние задания

1 ДЗ-11: YCrawler

Пишем асинхронный краулер для новостного сайта news.ycombinator.com

- | | | |
|-------|------------------------|--|
| 6 | Async. Часть 2 | async/await. Error handling |
| <hr/> | | |
| 7 | Golang. Часть 1 | Производительность Python. Golang. Toolchain, структура проекта, менеджмент зависимостей, тур по языку. |
| | | Домашние задания |
| | | 1 ДЗ-12: MemcLoad v2 |
| | | Создаем простого демона на Go, проводим сравнение с аналогичным на Python. |
| <hr/> | | |
| 8 | Golang. Часть 2 | Внутренности: горутины, сборщик мусора, оптимизации. |
| <hr/> | | |
| 9 | Profiling | Особенности архитектуры, характеристики железа. Антипаттерны профилирования. Методология. cProfile, line_profiler, memory_profiler. Инструменты Linux, perf. |
| <hr/> | | |
| 10 | Python 3 | Обзор изменений, новые фишки. Миграция проектов с 2 на 3 версию. |

5 Проектная работа

1 **Вводное
занятие**

2 **Консультация**

3 **Защита**