

O T U S

# Python Developer. Professional

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

Длительность курса: 146 академических часов

## 1 Advanced basics. Протоколы

### Цели занятия:

разобраться в особенностях представления естественных языков в Python;  
рассмотреть нюансы применения чисел с плавающей точкой;  
осознать систему протоколов языка;  
разобраться с концепцией итерирования в Python;  
получить представление о реализации и применимости итераторов.

### Краткое содержание:

изучим кодировки, Unicode в Python, coercion;  
рассмотрим числа с плавающей точкой, IEEE-754,  
особенности реализации, основные ошибки и особенности использования;  
рассмотрим итерирование, itertools, iterable, iterator, iterator protocol, sequence protocol.

### Домашние задания

#### 1 ДЗ-1: Log Analyzer

Цель: Цели и результат ДЗ искать в архиве с материалами занятия

Актуальное описание ДЗ искать в архиве с материалами занятия

---

2 **Advanced basics.**  
**"Граждане первого порядка"**

**Цели занятия:**

разобраться с особенностями применения ФП в Python; рассмотреть пространства имен и замыкания; рассмотреть устройство декораторов и способы их использования.

**Краткое содержание:**

изучим генераторы и корутины; рассмотрим функциональное программирование в Python, рекурсия и ее ограничения, tail recursion optimization; изучим особенности scoping, namespaces, замыкания и их устройство; рассмотрим декораторы функции и классы, декораторы с аргументами, вложенные декораторы.

---

3 **Internals.**  
**Виртуальная машина**

**Цели занятия:**

разобраться с устройством виртуальной машины; рассмотреть процесс исполнения кода; рассмотреть фундаментальные абстракции, которыми оперирует виртуальная машины.

**Краткое содержание:**

изучим виртуальную машину CPython, внутренности и архитектуру; исследуем байткод, опкоды; рассмотрим процесс интерпретации, основные типы данных виртуальной машины.

**Домашние задания**

1 ДЗ-2: CPython (опционально)

Цель: Цели и результат ДЗ искать в архиве с материалами занятия

Актуальное описание ДЗ искать в архиве с материалами занятия

---

4 **Internals.  
Управление  
памятью,  
"печально  
известный" GIL**

**Цели занятия:**

разобраться с работой основных типов данных и следствиями такой реализации;  
рассмотреть процесс управления памятью в Python;  
объяснить, как GIL влияет на производительность Python программ.

**Краткое содержание:**

изучим имплементацию основных типов данных и используемые оптимизации;  
рассмотрим систему управления памятью, reference counting, garbage collection;  
изучим GIL, реализация и следствия его существования, методы обхода.

---

5 **ООП. Объектная  
модель**

**Цели занятия:**

осознать устройство объектной модели Python;  
разобраться с разделением на новые и классические классы;  
объяснить тонкости множественного наследования;  
разобраться с нюансами реализации ООП в Python.

**Краткое содержание:**

рассмотрим объектную модель языка;  
изучим new-style классы, MRO, вызов super;  
исследуем множественное наследование;  
разберем slots, new, name mangling;  
рассмотрим интроспекцию.

**Домашние задания**

1 ДЗ-3.0: Scoring API

Цель: Цели и результат ДЗ искать в архиве с материалами занятия

Актуальное описание ДЗ искать в архиве с материалами занятия

---

## 6 ООР. Белая и черная "магия"

### Цели занятия:

объяснить дескрипторы и их протокол;  
разобраться с использованием "магических методов";  
рассмотреть область применимости абстрактных базовых классов;  
проанализировать особенности эксплуатации метаклассов.

### Краткое содержание:

рассмотрим дескрипторы, data и non-data дескрипторы, descriptor protocol, property;  
исследуем магические методы;  
рассмотрим модуль abc, абстрактные классы;  
рассмотрим метаклассы.

---

## 7 Testing. Дизайн тестов

### Цели занятия:

разобраться с конструированием кейсов тестирования;  
рассмотреть различия между видами тестирования;  
доказать необходимость тестирования и его место в жизненном цикле ПО.

### Краткое содержание:

определим необходимость тестирования;  
рассмотрим дизайн тестов и принципы подбора кейсов;  
исследуем различные виды тестирования;  
рассмотрим покрытие тестами, TDD.

### Домашние задания

#### 1 ДЗ-3.1: API Testing

Цель: Дописать тесты API

Дописываем тесты API

---

8 **Testing.**  
**Пирамида**  
**тестирования**

**Цели занятия:**

разобраться с устройством пирамиды тестирования;  
рассмотреть область применения инструментов  
тестирования (моков, фикстур и т.д.);  
разобраться с видами автоматизации тестирования.

**Краткое содержание:**

рассмотрим unit testing, integration testing;  
изучим антипаттерны тестирования;  
изучим mocking, monkey patching, фикстуры;  
исследуем вопрос автоматизации тестирования.  
Continuous integration, continuous delivery, continuous  
deployment.

---

9 **Automatization.**  
**Сетевое**  
**взаимодействие**

**Цели занятия:**

рассмотреть принципы сетевого взаимодействия через сокеты;  
разобраться с особенностями сетевых протоколов;  
рассмотреть нюансы написания программ, общающихся по сети.

**Краткое содержание:**

исследуем нюансы сетевого взаимодействия, socket;  
рассмотрим протокол TCP, реализацию клиента и сервера;  
рассмотрим протокол UDP, реализацию клиента и сервера;  
рассмотрим протокол HTTP и модель requests;  
изучим виды архитектуры web-server'ов.

**Домашние задания**

1 ДЗ-4: Web Server

Цель: Создать свой сервер и провести нагрузочное тестирование.

Создаем свой сервер на "ванильном" Python, частично реализующий протокол HTTP (будет корректно отдавать страницу wikipedia). Проводим нагрузочное тестирование.

---

10 **Automatization.  
Общение с БД и  
демонизация**

**Цели занятия:**

рассмотреть нюансы общения с серверной БД и основные паттерны;  
разобраться с процессом демонизации программ;  
разобраться с дистрибуцией Python программ.

**Краткое содержание:**

базы данных, коннекторы, ping-reconnect, конкурентное использование, connection pooling, "отстреливание" долгих запросов;  
кеширование;  
демонизация, конфиги, логирование;  
Setup tools и дистрибуция, менеджмент зависимостей.

## 1 Dynamic Web

### Цели занятия:

рассмотреть принципы функционирования динамического веба;  
разобраться с WSGI и его особенностями;  
поговорить о различных WSGI контейнерах.

### Краткое содержание:

CGI;  
FastCGI;  
WSGI;  
Gunicorn;  
uWSGI.

### 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.

---

### **Цели занятия:**

разобраться с классической структурой веб-сервисов;  
рассмотреть лучшие практики развертывания Django проектов;  
объяснить, как конфигурируются и эксплуатируются Django проекты.

### **Краткое содержание:**

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

---

### 3 Django. ORM и "зло"

#### Цели занятия:

поговорить о лучших практиках использования моделей;  
объяснить, как работает ORM и из чего он состоит;  
разобраться с написанием запросов через ORM;  
проанализировать запросы.

#### Краткое содержание:

модели;  
CRUD;  
ORM;  
Query sets, lazy evaluation;  
prefetch\preload, join'ы;  
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 layer

### Цели занятия:

объяснить, как навигироваться в комплексном пространстве мира распределенных систем и баз данных;  
познакомиться с основными понятиями, связанными с эксплуатацией и оптимизацией хранилища данных.

### Краткое содержание:

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 Django. Views

### Цели занятия:

разобраться с устройством логики представления в Django;  
объяснить, каким образом выбирать подход к описанию этой логики.

### Краткое содержание:

представления, виды представлений;  
function/class based views;  
generic views.

### Домашние задания

#### 1 ДЗ-6.1: Django project

Начинаем рисовать красивые странички

---

## 6 Django. Forms

### Цели занятия:

разобраться с логикой обработки форм и принципом их функционирования;  
объяснить, как устроен template engine и как это влияет на его производительность.

### Краткое содержание:

формы, поля;  
процесс валидации;  
model form. Widgets;  
templates, язык шаблонов;  
архитектура шаблонизатора;  
фильтры, тэги;  
Jinja2.

---

## 7 REST API. Know-how

### Цели занятия:

объяснить, что такое REST;  
разобраться с лучшими практиками реализации этого подхода.

### Краткое содержание:

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 **Web performance**

### **Цели занятия:**

рассмотреть пути масштабирования веб-проектов;  
разобраться с производительностью фронтенда.

### **Краткое содержание:**

Rate limiting;  
документация, swagger;  
Web Performance;  
фронтэнд оптимизация;  
масштабирование бекэнда.

## 1 Основы NumPy

### Цели занятия:

разобраться с областью применимости numpy и его основной структурой данных - ndarray;  
рассмотреть индексирование и операции над массивами;  
поговорить о возможностях библиотеки.

### Краткое содержание:

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 Advanced NumPy и IPython

### Цели занятия:

разобраться с внутренним устройством ndarray;  
рассмотреть продвинутые операции с массивами и broadcasting; познакомиться с основными возможностями ipython.

### Краткое содержание:

reshaping, broadcasting, structured и record массивы, хранение и загрузка данных; внутренности ndarray. Меммар, HDF5.

---

### 3 **Pandas. Машинное обучение и продакшен**

#### **Цели занятия:**

разобраться с областью применимости pandas и его основной структурой данных - dataframe;  
рассмотреть индексирование и операции над dataframe;  
поговорить о возможностях библиотеки;  
объяснить, как аналитические продукты интегрируются с продакшен системами.

#### **Краткое содержание:**

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 **Визуализация данных**

#### **Цели занятия:**

разобраться с видовым многообразием библиотек визуализации данных в Python;  
рассмотреть основные возможности matplotlib и два его интерфейса;  
рассмотреть основные возможности seaborn и понять его преимущества над matplotlib;  
познакомиться с принципами визуализации данных.

#### **Краткое содержание:**

графики, гистограммы, subplots, аннотации, стили;  
data visualization, принципы правильных визуализаций.

## 1 Concurrency. Потоки

### Цели занятия:

разобраться с терминологией конкурентного программирования;  
рассмотреть ограничения, накладываемые виртуальной машиной на multithreading;  
познакомиться с примитивами синхронизации из стандартной библиотеки.

### Краткое содержание:

concurrency, parallelism;  
IO/CPU bound задачи;  
Multithreading, sharing, communication.

### Домашние задания

#### 1 ДЗ-9: MemcLoad

Реализуем конкурентную заливку данных в memcache'ы

---

## 2 Concurrency. Процессы

### Цели занятия:

разобраться с превратностями запуска процессов в разных ОС;  
познакомиться с возможностями коммуникации процессов для выполнения общей задачи;  
рассмотреть устройство основных примитивов синхронизации.

### Краткое содержание:

multiprocessing, IPC, shared memory, Manager;  
Distributed computing.

---

### 3 C extensions

#### Цели занятия:

разобраться с основными принципами написания расширений на языке C;  
познакомиться с управлением памятью через reference counting;  
рассмотреть подход к созданию сложных объектов.

#### Краткое содержание:

расширение 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

#### Цели занятия:

разобраться с генерацией C расширений из Python кода с помощью Cython;  
познакомиться с синтаксисом Cython;  
рассмотреть область применимости PyPy;  
научиться вызывать функции из уже скомпилированных приложений через ffi и ctypes/.

#### Краткое содержание:

ffi;  
Cython;  
PyPy.

---

## 5 Asyncio. Origins

### Цели занятия:

познакомиться с историей появления asyncio в Python;  
разобраться с коррутинами и yield from;  
разобраться с futures.

### Краткое содержание:

generators, coroutines, yield from;  
event loop;  
future.

### Домашние задания

#### 1 ДЗ-11: YCrawler

Пишем асинхронный краулер для новостного сайта  
news.ycombinator.com

---

## 6 Asyncio. Event loop, async/await, low/high level API

### Цели занятия:

познакомиться с концепцией event loop, старым и  
новым синтаксисом асинхронных вызовов;  
рассмотреть высоко- и низкоуровневое API asyncio;  
разобраться с внутренним устройством awaitable  
объектов.

### Краткое содержание:

async/await;  
error handling.

---

## 7 Golang. Тур по языку

### Цели занятия:

познакомиться с синтаксисом, основными идиомами и экосистемой языка;  
проанализировать область применимости языка, его сильные и слабые стороны;  
провести сравнение с Python.

### Краткое содержание:

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.

---

## 8 Golang. Особенности внутреннего устройства

### Цели занятия:

познакомиться с особенностями реализации языка;  
рассмотреть принципы работы с памятью;  
разобраться с диспетчером горутин;  
познакомиться с примерами реальных задач, решаемых с помощью Go.

### Краткое содержание:

внутренности: горутины, сборщик мусора, оптимизации.

---

## 9 Profiling

### Цели занятия:

проанализировать особенности архитектуры,  
характеристики железа;  
познакомиться с методиками профилирования кода;  
разобраться с профилированием памяти и ЦПУ;  
познакомиться с инструментами для профилирования,  
предоставляемыми в linux.

### Краткое содержание:

антипаттерны профилирования;  
методология. cProfile, line\_profiler, memory\_profiler;  
инструменты Linux, perf.

---

## 10 Python 2 vs Python 3

### Цели занятия:

рассмотреть особенности предыдущей версии;  
обсудить миграцию проектов с 2 на 3 версию.

### Краткое содержание:

причины возникновения проблемы, наличия двух  
используемых версий языка;  
процесс миграции со 2 на 3 версию.

## 1 Выбор темы и организация проектной работы

### Цели занятия:

выбрать и обсудить тему проектной работы;  
спланировать работу над проектом;  
ознакомиться с регламентом работы над проектом.

### Краткое содержание:

правила работы над проектом и специфика проведения итоговой защиты;  
требования к результату проекта и итоговой документации.

### Домашние задания

#### 1 Проект

Цель: - закрепить знания и навыки, полученные в течение курса;

- реализовать собственный проект;
- пополнить своё портфолио качественным проектом.

- выбрать тему;
- подтвердить тему в чате с преподавателем;
- предоставить промежуточные наработки на промежуточное ревью;
- предоставить финальную версию проекта к дедлайну.

---

## 2 Консультация по проектам и домашним заданиям

### Цели занятия:

получить ответы на вопросы по проекту, ДЗ и по курсу.

### Краткое содержание:

вопросы по улучшению и оптимизации работы над проектом;  
затруднения при выполнении ДЗ;  
вопросы по программе.

---

**3 Защита  
проектных  
работ**

**Цели занятия:**

защитить проект и получить рекомендации экспертов.

**Краткое содержание:**

презентация проектов перед комиссией;  
вопросы и комментарии по проектам.