

Тема

Введение в Python



Николай Пекальн

Director of analytics Vezet

- 2011-2018 прошёл путь от аналитика до зам. коммерческого директора по аналитике в Wildberries
- 2018 директор по аналитике Юла в Mail.ru Group
- с 2019 директор по аналитике и машинному обучению в Везёт

Содержание урока

- ❑ История Python
- ❑ Python 2 и Python 3
- ❑ Языки программирования и конкретно Python
- ❑ Особенности Python
- ❑ Примеры использования Python в аналитике и не только
- ❑ Какими инструментами будем пользоваться?
- ❑ Командная строка и установка интерпретатора
- ❑ Пример работы с Jupyter и первый код на Python
- ❑ Что такое PEP-8?

История Python

Создание Python



Гвидо Ван Россум

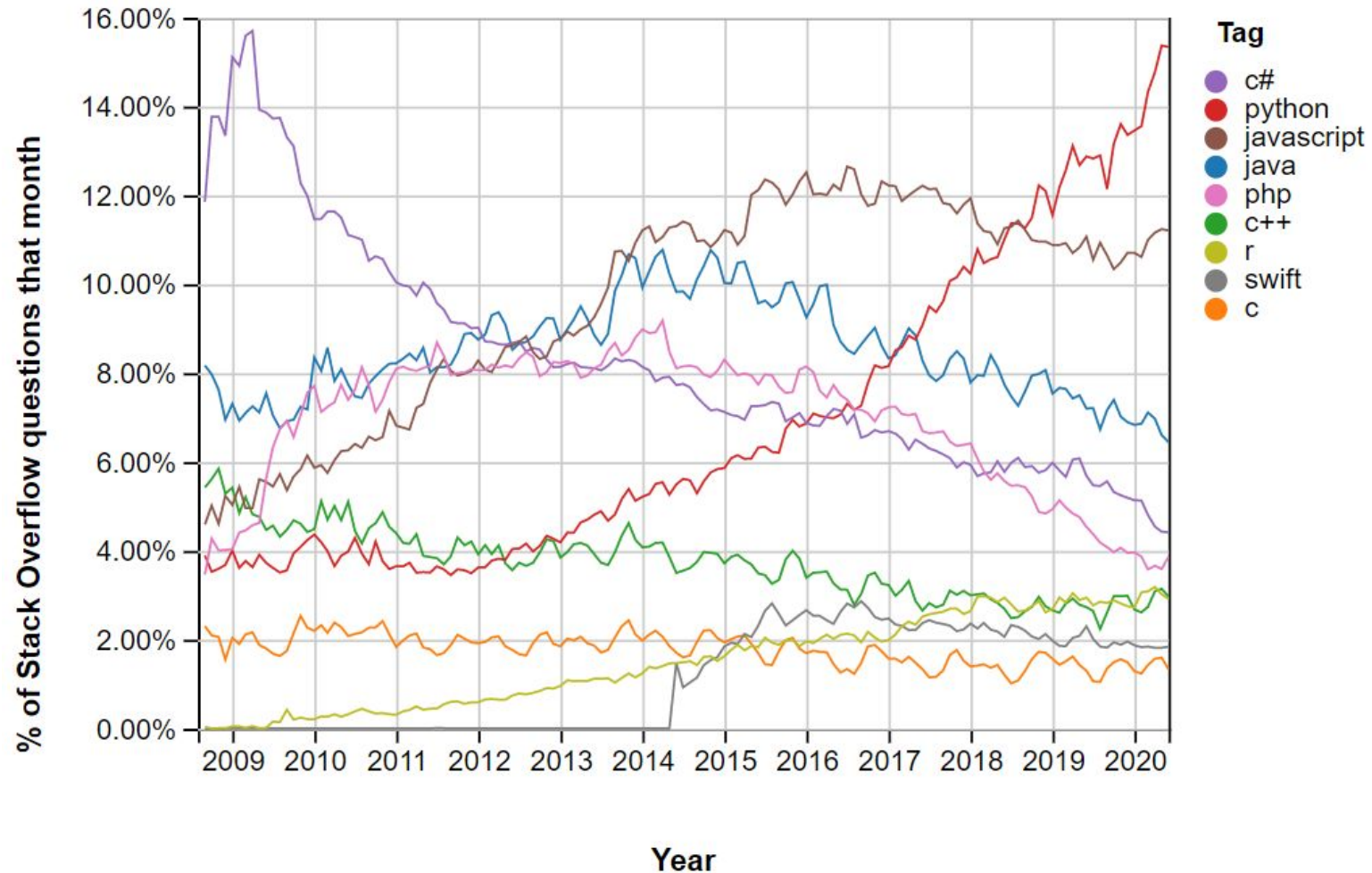
Создатель языка Python

- ❑ Первые наработки — конец 80-х
- ❑ Python 1.0 — Январь 1994
- ❑ Python 2.0 — Октябрь 2000
- ❑ Python 3.0 — Декабрь 2008

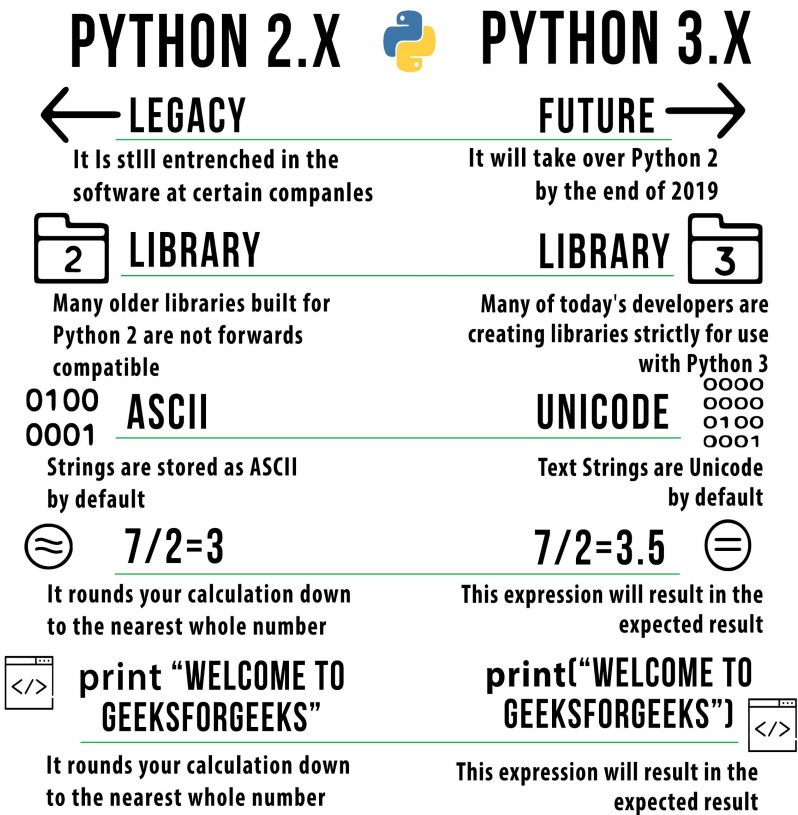
Zen of Python

- ❑ Красивое лучше, чем уродливое.
- ❑ Явное лучше, чем неявное.
- ❑ Простое лучше, чем сложное.
- ❑ Сложное лучше, чем запутанное.
- ❑ Плоское лучше, чем вложенное.
- ❑ Разреженное лучше, чем плотное.
- ❑ Читаемость имеет значение.
- ❑ Особые случаи не настолько особые, чтобы нарушать правила.
- ❑ При этом практичность важнее безупречности.
- ❑ Ошибки никогда не должны замалчиваться.
- ❑ Если не замалчиваются явно.
- ❑ Встретив двусмысленность, отбрось искушение угадать.
- ❑ Должен существовать один — и, желательно, только один — очевидный способ сделать это.
- ❑ Хотя он поначалу может быть и не очевиден, если вы не голландец.
- ❑ Сейчас лучше, чем никогда.
- ❑ Хотя никогда зачастую лучше, чем прямо сейчас.
- ❑ Если реализацию сложно объяснить — идея плоха.
- ❑ Если реализацию легко объяснить — идея, возможно, хороша.
- ❑ Пространства имён — отличная вещь! Давайте будем делать их больше!

Популярность Python



Python 2 и Python 3



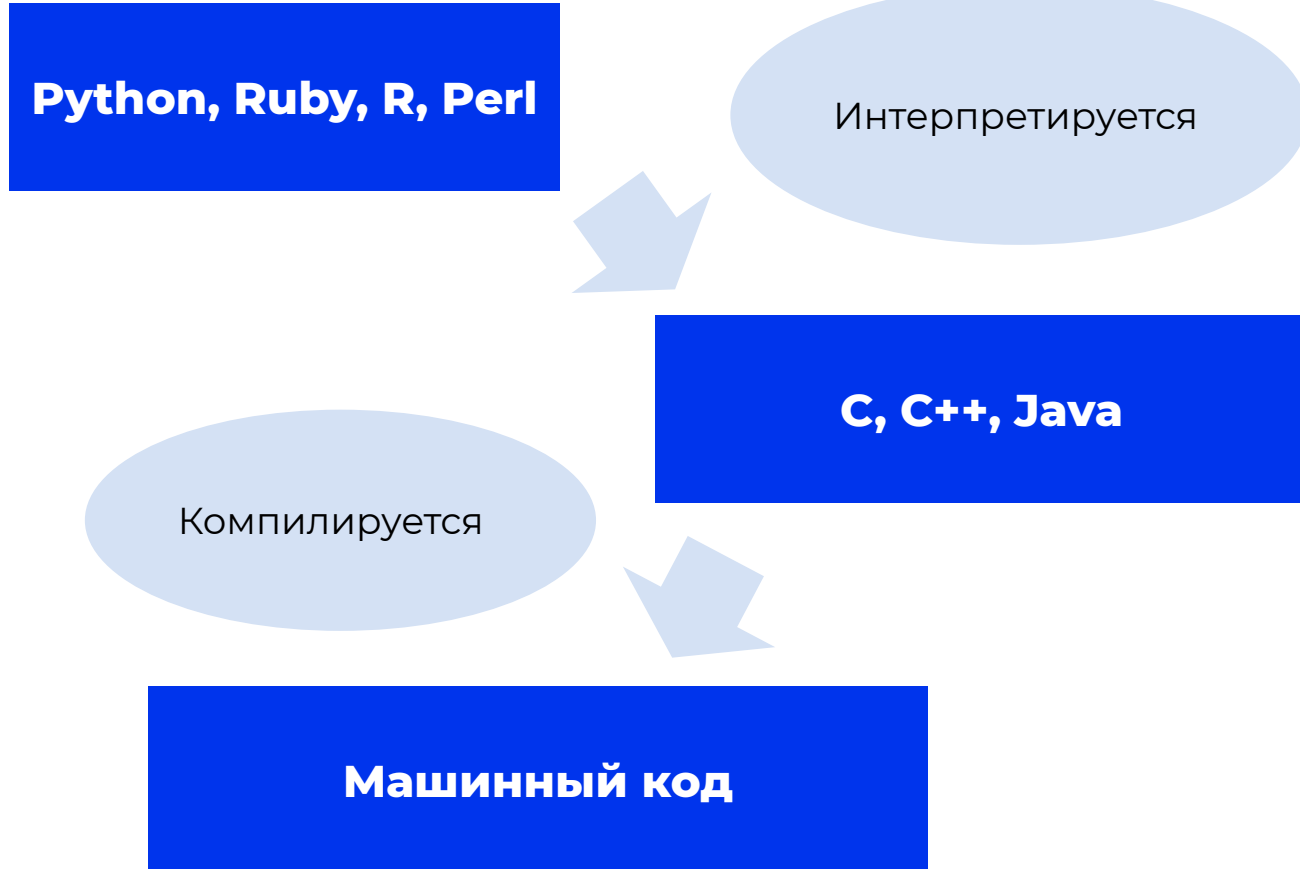
На курсе мы будем работать с Python 3.

С апреля 2020 года Python 2 перестал развиваться.

Однако важно помнить, что код написанный на Python 2 может не запуститься интерпретатором Python 3 и наоборот.

Языки программирования и конкретно Python

Python и другие языки программирования



Основные дистрибутивы Python



CPython

Основная реализация Python на языке C.



Jython

Реализация Python на Java.
На данный момент не пользуется большой популярностью.

IronPython

IronPython

Реализация Python под Microsoft.NET на языке C#.



PyPy (PythonPython)

JIT реализация через на RPython



PyMite

Реализация Python для ряда микропроцессоров (Atmel и др.)

Главные особенности Python?

#1

**Язык программирования
общего назначения**

Можно написать что угодно.

#2

Простой

Если что-то написано непонятно,
значит написано плохо.

#3

Интерпретируемый

Код — текстовые файлы.
Но может под тормаживать.

#4

Популярный

Куча библиотек для чего угодно
и хелпов с туториалами.

Технические особенности Python?

#5

Мультиплатформенный

Есть реализации под почти любое устройство.

#6

Мультипарадигмальный

Умеет и функции и объекты.

#7

С динамической типизацией

Типы определяются на лету.

#8

С полной интроспекцией

Программе доступен собственный код.

Примеры использования Python в аналитике и не только

Задачи аналитики, решаемые на Python

#1

Анализ данных:

Рассчитать статистики по большому объему данных, подсчитать итоги A/B теста.

#3

Machine Learning:

Распознавание картинок и рекомендации покупок.

#5

Web - разработка

Небольшие веб-сервисы и сайтики для отчётности и не только.

#2

Визуализация данных:

Отобразить зависимости одних данных от других, построить графики и диаграммы.

#4

ETL-разработка

Загрузка данных из разных источников, приведение их к нужному виду, загрузка в аналитическое хранилище.

Задачи аналитики, решаемые на Python

#1

Анализ данных:

Рассчитать статистики по большому объему данных, подсчитать итоги A/B теста.

#2

Визуализация данных:

Отобразить зависимости одних данных от других, построить графики и диаграммы.

#3

Machine Learning:

Распознавание картинок и рекомендации покупок.

#4

ETL-разработка

Загрузка данных из разных источников, приведение их к нужному виду, загрузка в аналитическое хранилище.

#5

Web - разработка

Небольшие веб-сервисы и сайтики для отчётности и не только.

Задачи аналитики, решаемые на Python

#1

Анализ данных:

Рассчитать статистики по большому объему данных, подсчитать итоги A/B теста.

#3

Machine Learning:

Распознавание картинок и рекомендации покупок.

#5

Web - разработка

Небольшие веб-сервисы и сайтики для отчётности и не только.

#2

Визуализация данных:

Отобразить зависимости одних данных от других, построить графики и диаграммы.

#4

ETL-разработка

Загрузка данных из разных источников, приведение их к нужному виду, загрузка в аналитическое хранилище.

Задачи аналитики, решаемые на Python

#1

Анализ данных:

Рассчитать статистики по большому объему данных, подсчитать итоги A/B теста.

#3

Machine Learning:

Распознавание картинок и рекомендации покупок.

#5

Web - разработка

Небольшие веб-сервисы и сайтики для отчётности и не только.

#2

Визуализация данных:

Отобразить зависимости одних данных от других, построить графики и диаграммы.

#4

ETL-разработка

Загрузка данных из разных источников, приведение их к нужному виду, загрузка в аналитическое хранилище.

Задачи аналитики, решаемые на Python

#1

Анализ данных:

Рассчитать статистики по большому объему данных, подсчитать итоги A/B теста.

#3

Machine Learning:

Распознавание картинок и рекомендации покупок.

#5

Web - разработка

Небольшие веб-сервисы и сайтики для отчётности и не только.

#2

Визуализация данных:

Отобразить зависимости одних данных от других, построить графики и диаграммы.

#4

ETL-разработка

Загрузка данных из разных источников, приведение их к нужному виду, загрузка в аналитическое хранилище.

**Какими инструментами
будем пользоваться?**

Где и как писать код на Python?

#1

**Текстовый редактор, блокнот.
Файлы с расширением .py**

Писать код можно в любом текстовом редакторе.

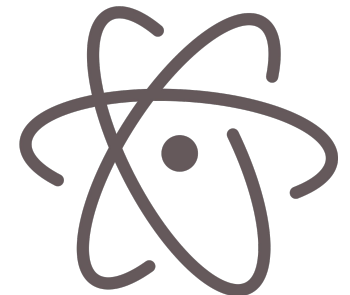
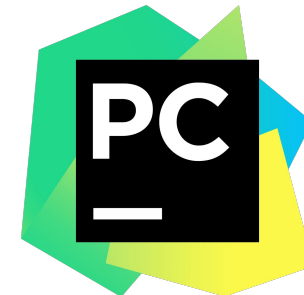


Где и как писать код на Python?

#2

IDE. Специализированные среды для разработки

Подсветка синтаксиса, возможно подключения дополнительных модулей. IDE создаются, чтобы сделать жизнь разработчика проще.



Где и как писать код на Python?

#3

Notebooks

Рабочие тетради с возможностью поблочного выполнения кода и оформления заметок.



Все что нужно, чтобы писать на Python



**Интерпретатор
Python 3.4 и старше**



**Терминал /
Командная строка**



**Jupyter
Notebook**

Как установить?

#1

Установить интерпретатор
с <https://www.python.org/>

#2

Установить JupyterLab
через pip

Пример работы с Jupyter и первый код на Python

Что такое РЕР-8?

Правила написания кода

Плохой код

```
a=2;b      =7  
print      (a +b)  
c          =a+b;print(a*c)
```

Хороший код

```
a = 2  
b = 7  
print(a + b)  
c = a + b  
print(a * c)
```