

Введение

ERAM, история и особенности Python, инструменты разработчика



- Компания основана в 1993 году
- Программные решения для клиентов по всему миру в самых разных областях (от медиа до биотехнологий)
- 25+ стран, больше 25 тысяч сотрудников

Содержание

1. Введение
2. Типы и структуры данных
3. Функции – 1 (определение, области видимости, args, kwargs...)
4. Итераторы, генераторы
5. Функции – 2 (рекурсия, lambda, декораторы)
6. Функциональное программирование (парадигма f-программирования, filter, map, reduce, functools)
7. Классы и ООП (классы, наследование, MRO, __dict__, __slots__)
8. Классы и ООП – 2 (декрипторы, метаклассы, абстрактные классы, classmethod, staticmethod)
9. Исключения и контекст-менеджеры
10. Потоки (GIL, потоки и процессы)
11. Сокеты (сокеты, UDP, TCP)
12. Анализ текстов
13. Анализ данных
14. DB (библиотеки и фреймворки для работы с базами данных)
15. Python & WEB – 1 (фреймворки для web-разработки (Flask, Django, Bottle))
16. Python & WEB – 2 (приложение на Flask))
17. Тестирование и покрытие (Unititest, Pytest, Coverage, Mocks, Selenium)
18. Экзамен

Гвидо ван Россум



Создатель (1991 г.) и
пожизненный великодушный
диктатор до 2018 года

Интерпретируемый

Код

```
def increment(variable):  
    return variable + 1
```



2

```
0 LOAD_FAST  
2 LOAD_CONST  
4 BINARY_ADD  
6 RETURN_VALUE
```

Байт-код

```
0 (variable)  
1 (1)
```

Динамический

```
In [5]: increment('string')
```

```
-----  
TypeError                                Traceback (most recent call last)  
<ipython-input-5-4fb94858eab2> in <module>()  
----> 1 increment('string')  
  
<ipython-input-2-25bec8277a7f> in increment(variable)  
      1 def increment(variable):  
----> 2     return variable + 1  
  
TypeError: must be str, not int
```

С интроспекцией

```
In [17]: def decrement(variable):  
         return variable - 1
```

```
In [18]: decrement.__code__ = increment.__code__
```

```
In [19]: decrement(5)
```

```
Out[19]: 6
```

А также:

- Множество реализаций (**CPython**, Jython, PyPy)
- Поддерживает много парадигм (ООП и не только)
- Лаконичный синтаксис
- Широкое поле применения (разработка, анализ данных, dev-ops)

Python 2 vs. Python 3

- Python 3 появился в 2008 году
- Python 2 – поддерживается, Python 3 – развивается.
- Имеют важные отличия:
 - работа со строками
 - иерархия классов
 - разная реализация многих стандартных методов и др.
- Будьте внимательны при чтении документации – это про второй или третий питон??!!
- Мы будем говорить о Python 3

Как и в чем писать код?

Что нужно чтобы писать код?

- Редактирование кода
- Запуск
- Отладка
- Упорядочивание и хранение множества файлов\директорий, поиск по ним
- Работа с системами контроля версий

Как и где писать код?

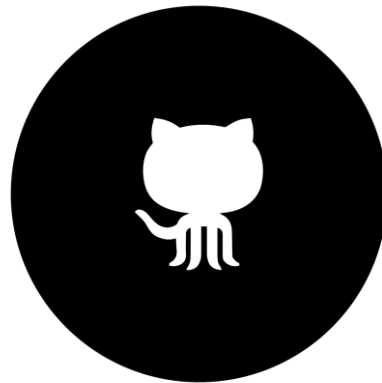
Это выбор каждого!

- Просто редактор + один или несколько терминалов
 - Удобнее редактировать/писать/читать код
 - Больше свободы, меньше «хлама»
- Интегрированная среда разработки
 - Все сразу под рукой
 - Множество фиц (подсказка сигнатур, автопереименование и прочее)

Также возможны – GUI для работы с системами контроля версий

С чем есть git?

- Репозиторий – множество папок и файлов, имеющее имя, чья история изменений фиксируется, ветвится, синхронизируется между разными участниками
- Бесплатные сервисы для хранения и передачи кода - Gitlab, Github, BitBucket



С чем есть git?

В терминале:

git clone <https://gitlab.com/cheburashka/super-cool.git>

- клонирует репозиторий к вам

git add my_code.py (а лучше git add -i)

- индексирует то, что вы хотите внести в качестве изменений

git commit -m "describe what the changes are about in this message"

- собственно фиксирует изменения, сопровождая пояснением, и порождает новую точку в развитии кода

git push

- вуаля, ваши изменения синхронизируются

git pull

- мы делаем clone или pull вашего репозитория и радуемся

Домашнее задание

- Поставить Python 3 (<https://www.python.org/downloads/>)
- Выбрать редактор/IDE и установить (PyCharm, Sublime Text, Atom - что хотите)
- Поставить себе git (<https://git-scm.com/downloads>)
- Создать открытый для всех репозиторий (если у вас такой уже есть – замечательно)