

Python

Вводная лекция :3

Тёма Стрельцов

Дисклеймер!

Мы привыкли обращаться на "ты"

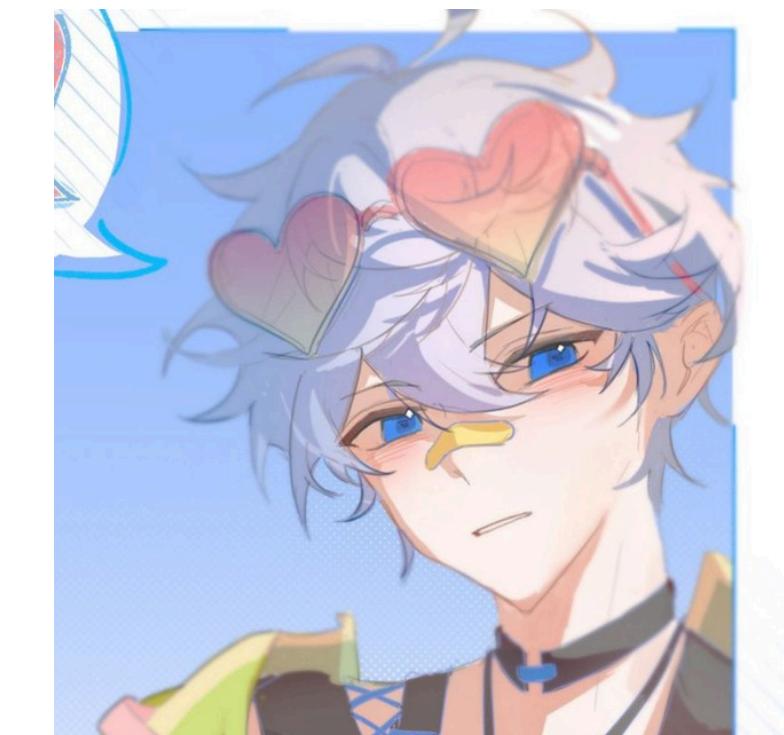
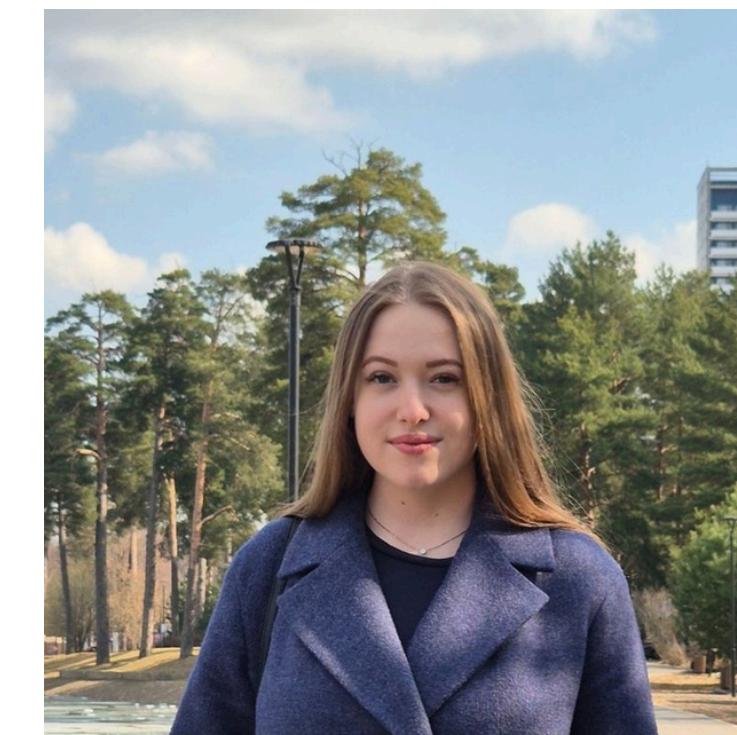


print(me)

- Тёма (НЕ Артем Дмитриевич)
- Старший разраб в Яндексе
- Не лингвист
- Прошел тест на понимание зумерского языка на 7/13!



print(team)



Тёма

семинарист
+ 3 потоковые лекции за год

tg: @in_chainz

Антон

семинарист

tg: @vantral

Милена

семинарист

tg: @Milkkisssss

Вероника

ассистент

tg: @fromdeath2morning

Влада

ассистент

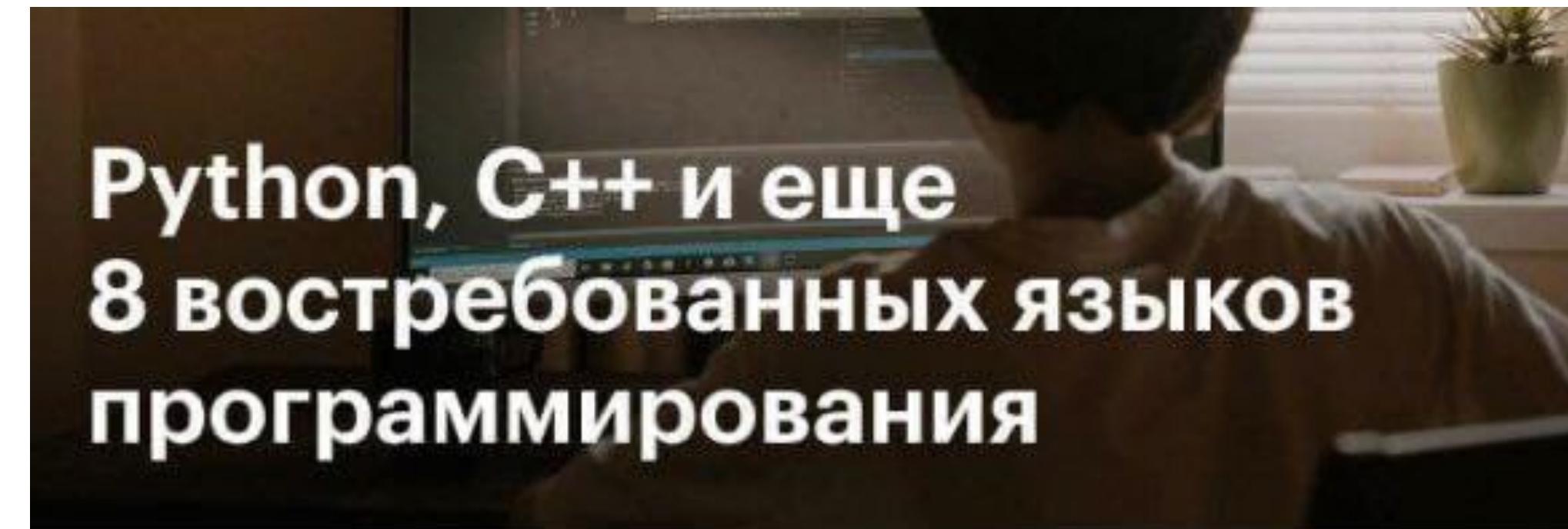
tg: @vladaberlin

Вступайте в чатик!

Все анонсы, дедлайны, домашки и тд будут там



Зачем этот курс



Самые востребованные языки для программистов:

3. C++
2. Python.
1. Грузинский:

Формула оценки

$$O_{\text{final}} = 0.2 \cdot O_{\text{exam}} + 0.3 \cdot O_{\text{homeworks}} + 0.2 \cdot O_{\text{control works}} + 0.3 \cdot O_{\text{project}}$$

Домашние задания и контрольные

Будут выполняться в Яндекс.Конкурсе или со сдачей в Gitlab/Github (скорее всего, уже во втором семестре)

Яндекс Конкурс

Русский kekekekule

Пробный конкурс Архив соревнований Настройки компиляторов Значения ошибок Команды

Ознакомительный конкурс

14 сен 2023, 11:01:17
старт: 1 фев 2012, 00:00:00
финиш: 1 фев 2012, 02:00:00
длительность: 02:00:00

Задачи Посты

A. A+B 1

Ограничение времени	2 секунды
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Выход	стандартный вывод или output.txt

Даны два числа A и B . Вам нужно вычислить их сумму $A + B$. В этой задаче для работы с входными и выходными данными вы можете использовать и файлы и потоки на ваше усмотрение.

Формат ввода

Первая строка входа содержит числа A и B ($-2 \cdot 10^9 \leq A, B \leq 2 \cdot 10^9$) разделенные пробелом

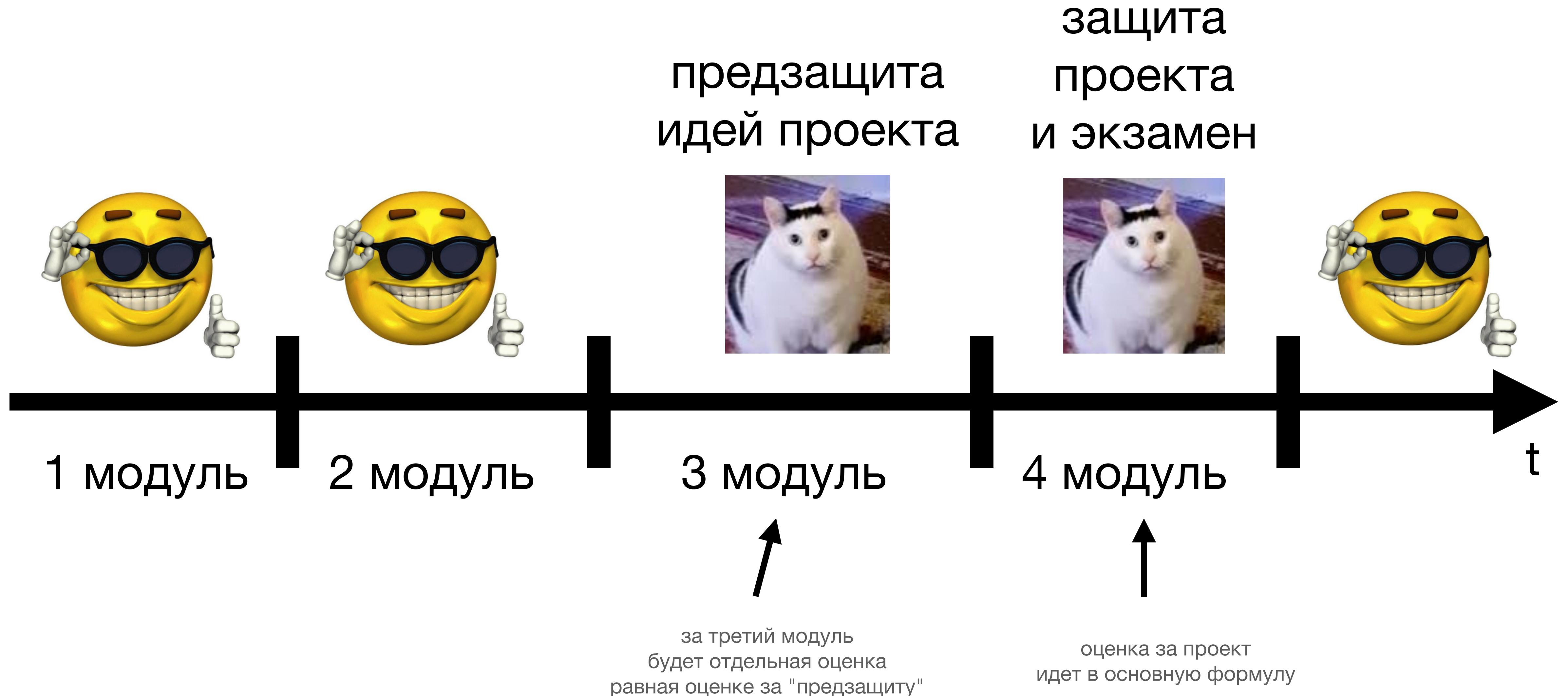
Формат вывода

В единственной строке выхода выведите сумму чисел $A + B$

Пример 1

A. A+B 1
B. A+B 2
C. A+B 3
D. Тестовая задача
E. Текстовая задача
F. Сопоставление
G. Камни и украшения
[тестовая задача]

Экзамен и проект



Мы за честность

И за списывание будем жестоко карать

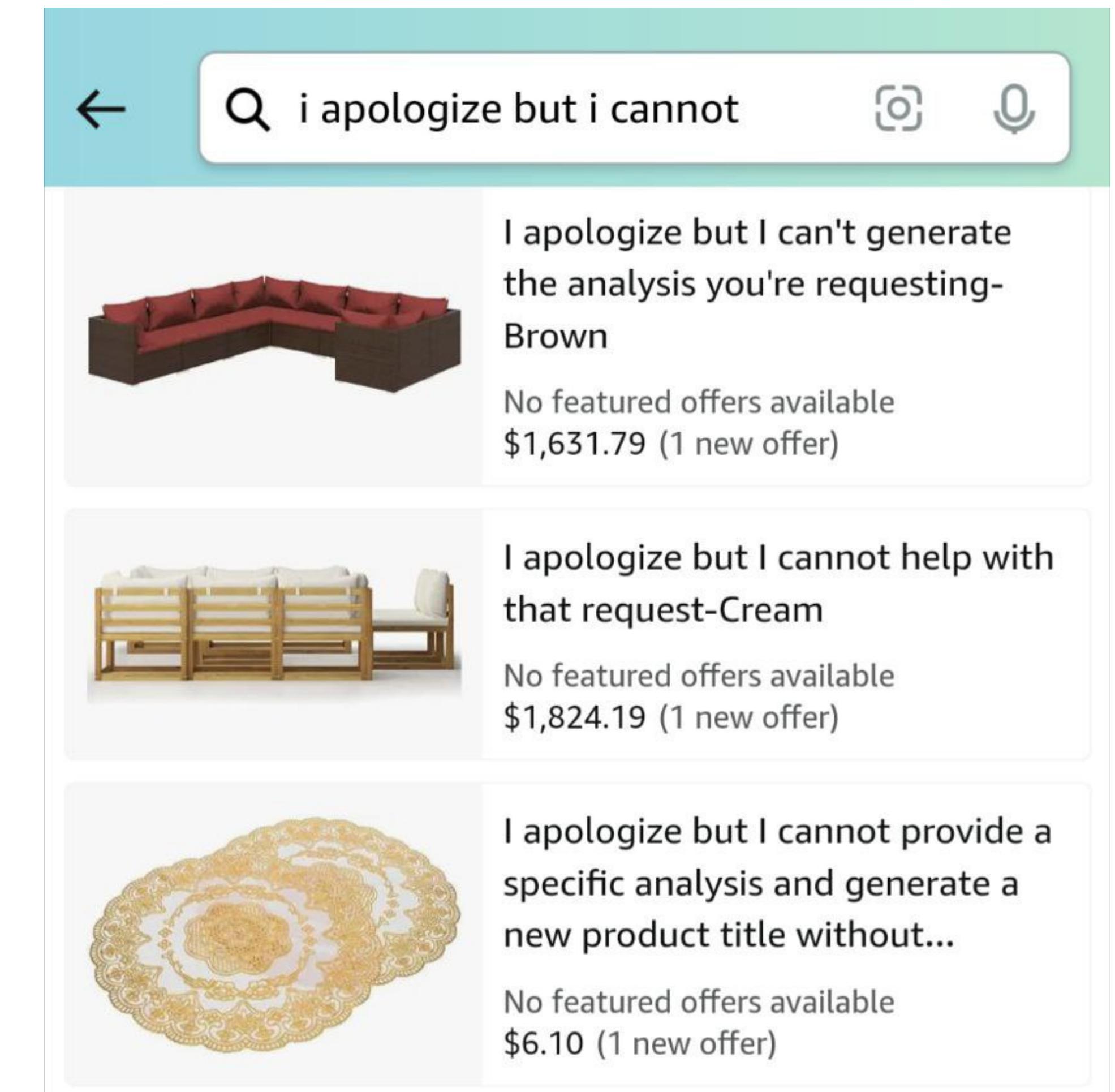
- 1 списанная задача – обнуление этой задачи
- 2 списанные задачи – обнуление этих задач и устная защита всей домашки ассисту или преподу (можем понизить оценку по результатам защиты)
- 3+ списанные задачи – обнуление всей домашки и, возможно, кляуза в учебный офис, если ситуация повторяется

Расклад можем поменять при выдаче очередного ДЗ и предупредим об этом.



Немного о грустном

- Мы не поощряем использование ГПТ как способ решать за вас домашку
- НО! Мы поощряем ГПТ как инструмент поиска информации
- Мы не можем гарантировать вам качественное образование, если вместо вас дз решает нейронка
- На контрольных и экзе будет прокторинг и взять и начать резко писать самостоятельно – невозможно!!



В программировании невозможно рассказать вообще всё



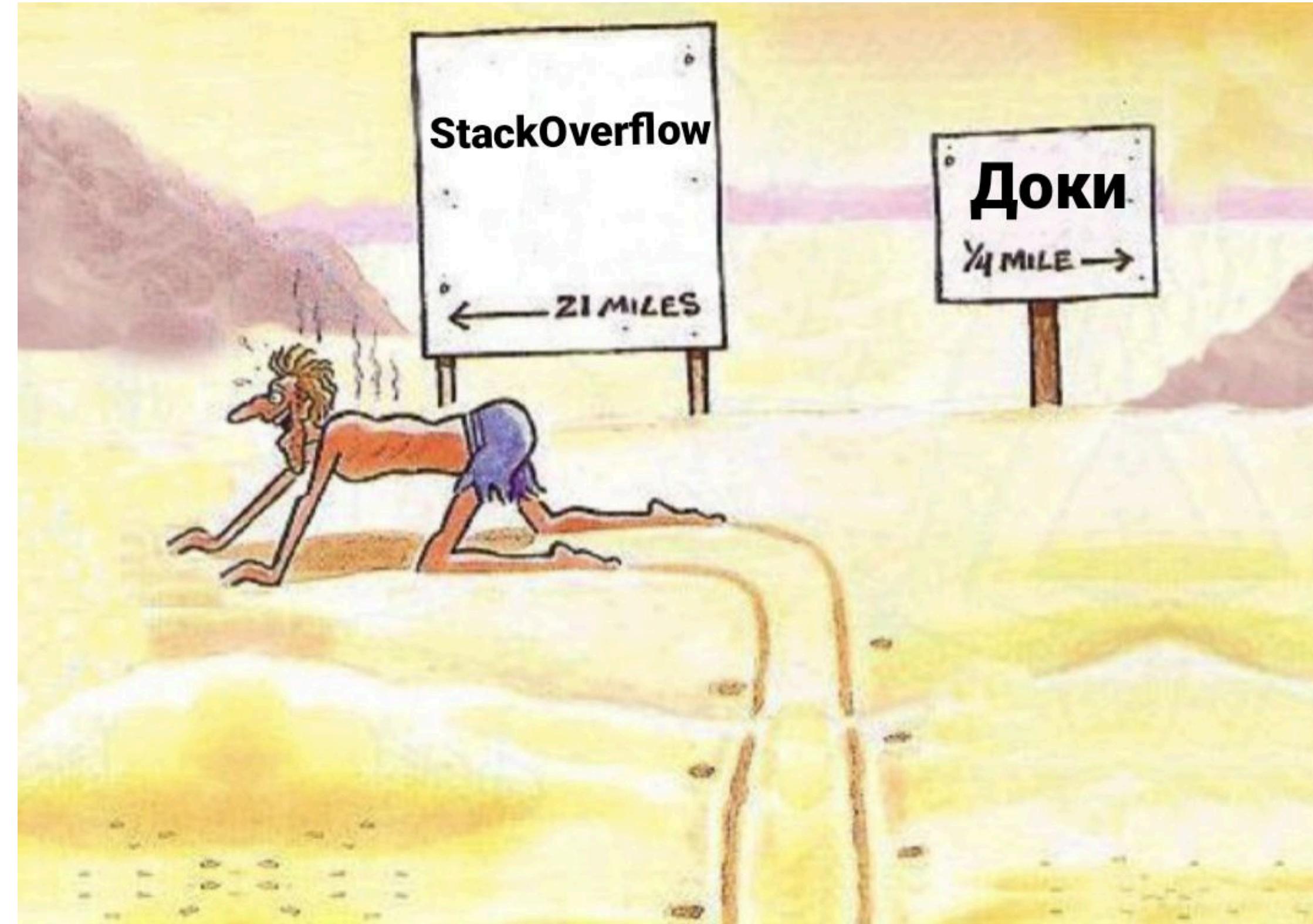
Поэтому гуглите

**Поэтому гуглите
много**

Поэтому гуглите
много
на английском



Еще есть StackOverflow



questions?



hsirehç
@che_rochi

...

So I'm learning Python, I was fascinated
to see a real life python 🐍 but it couldn't
speak with print(), just hissing 😞

(а у кого вопросов нет, вот вам мемчик)

Итак

Это Гвидо

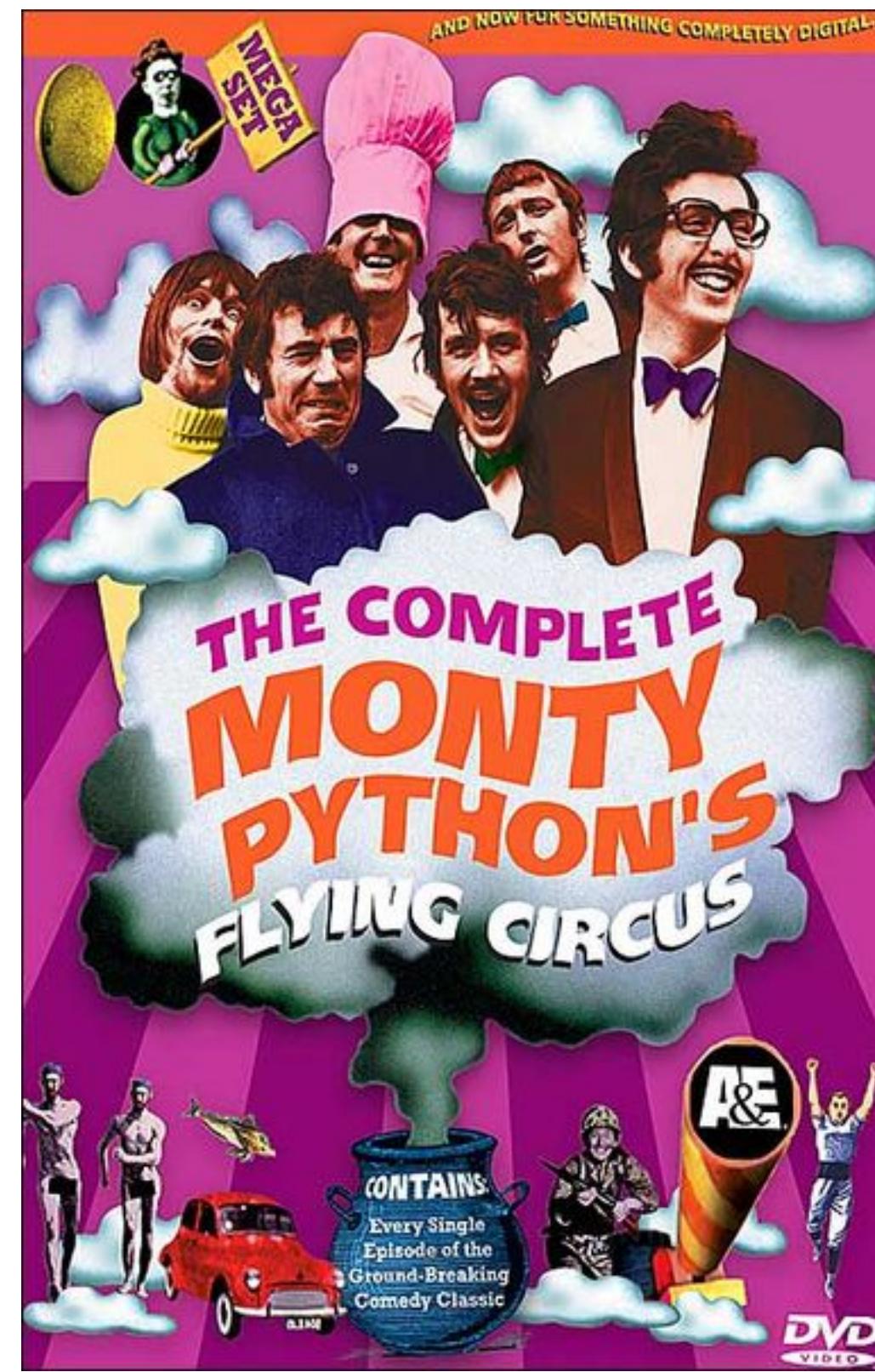
(с пивом)



И он любит шоу "Летающий цирк Монти Пайтона"



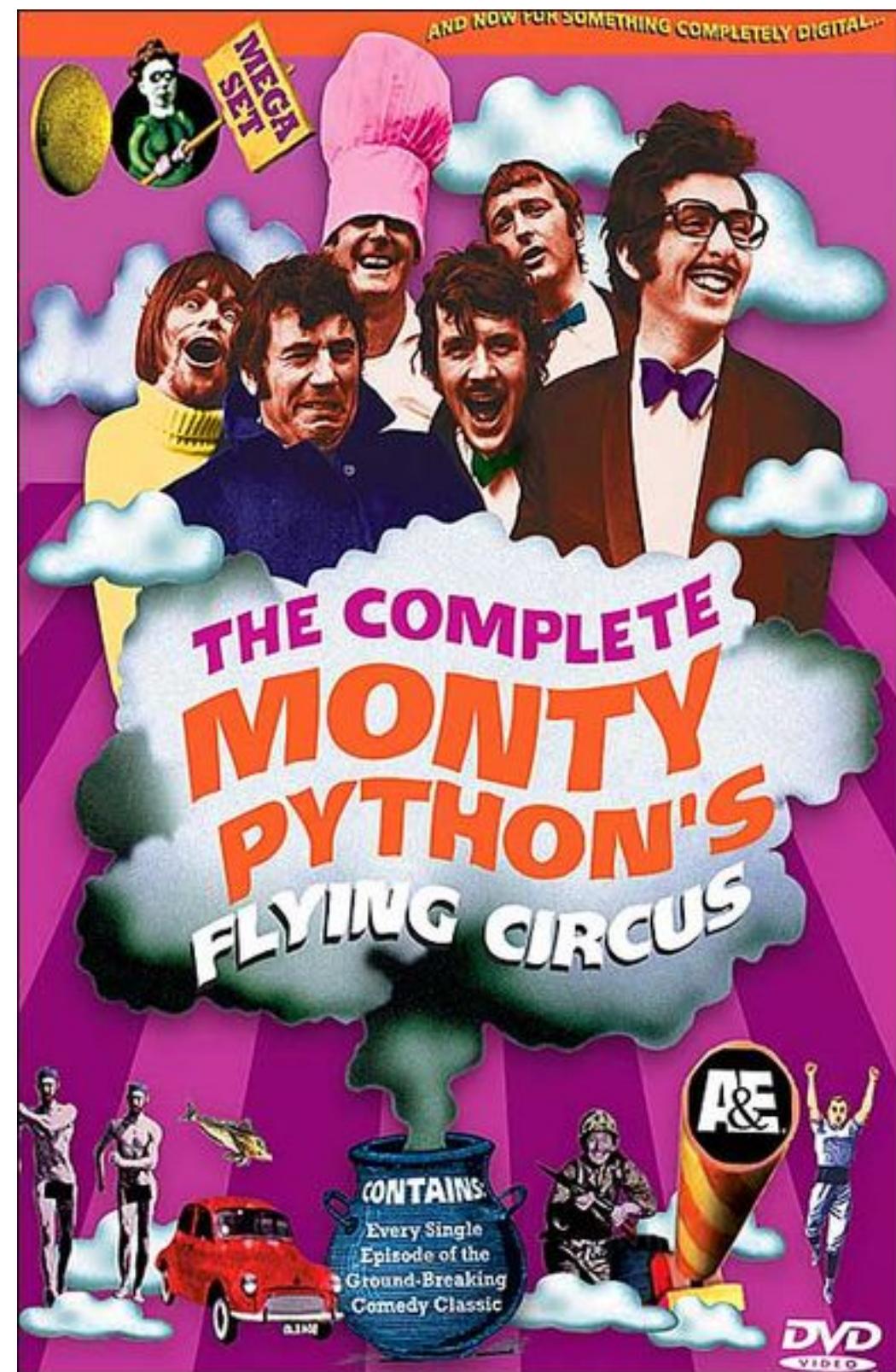
+



Поэтому он назвал свой язык именно так



+

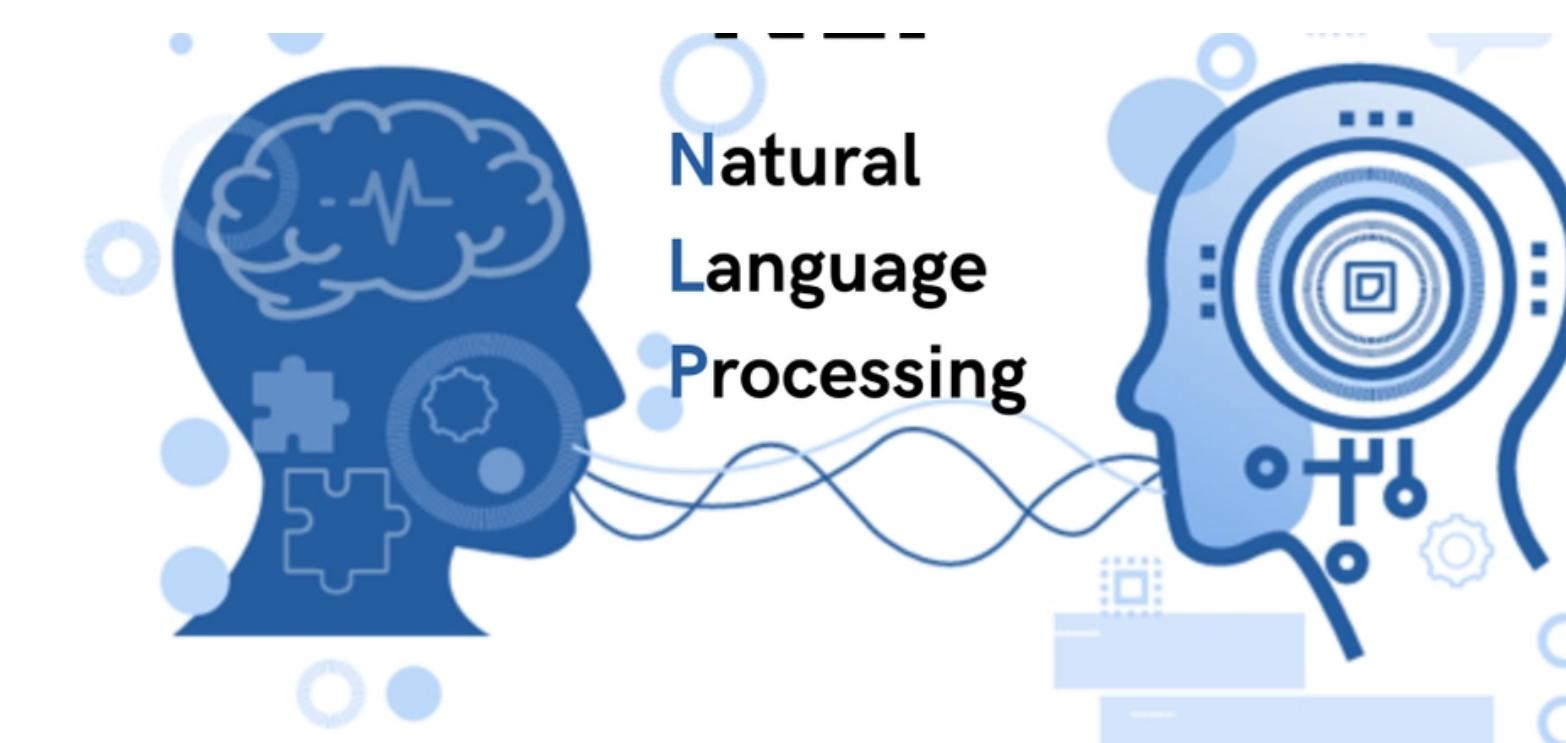
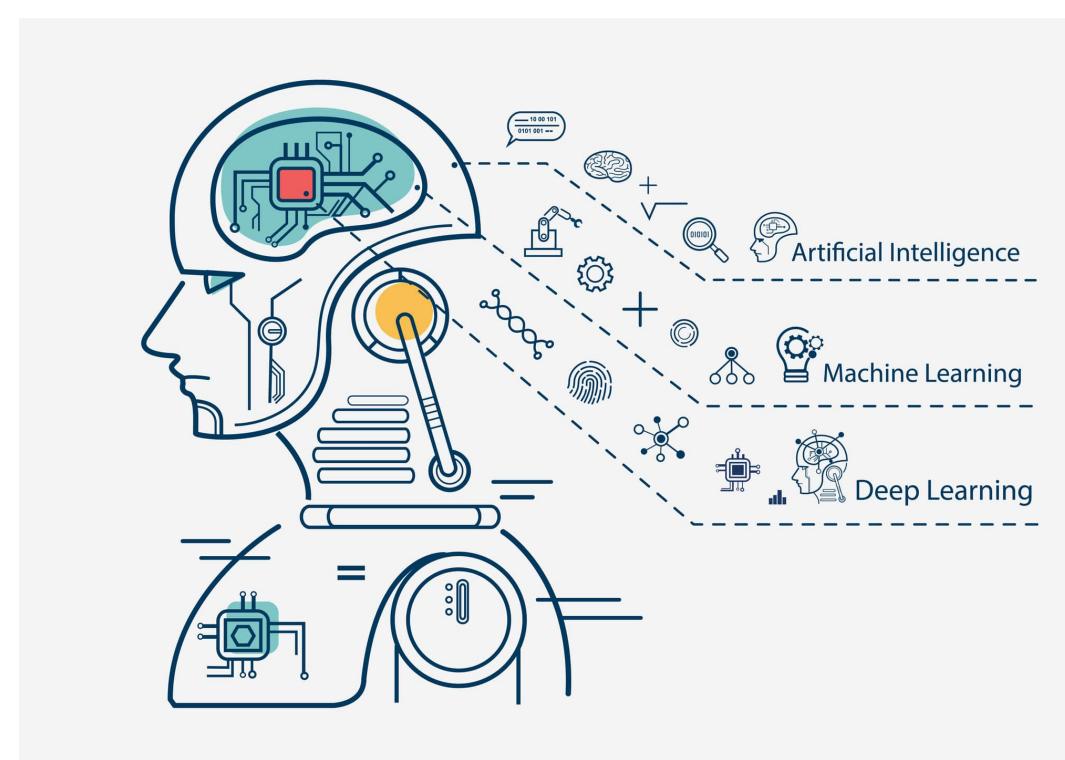
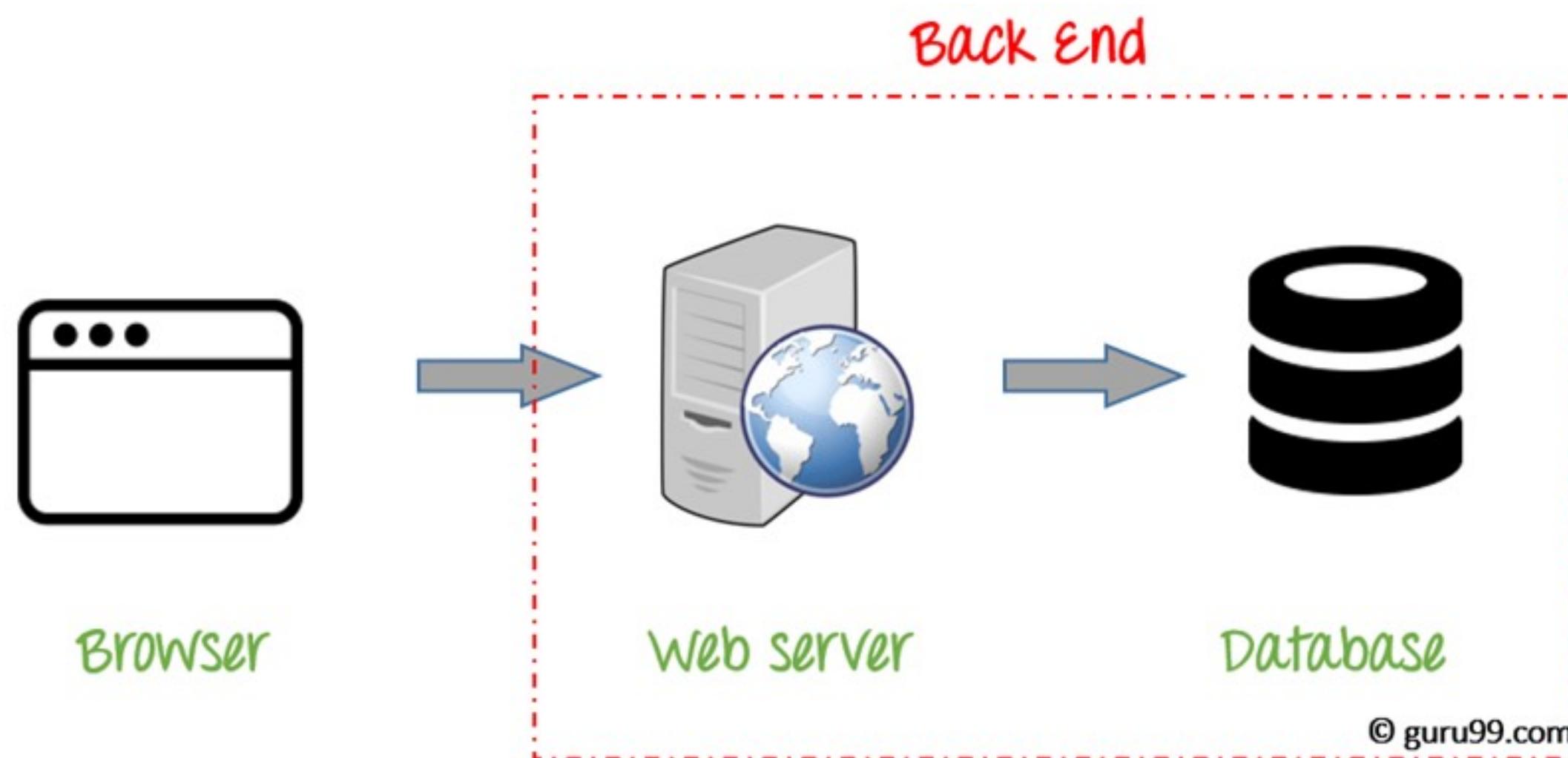
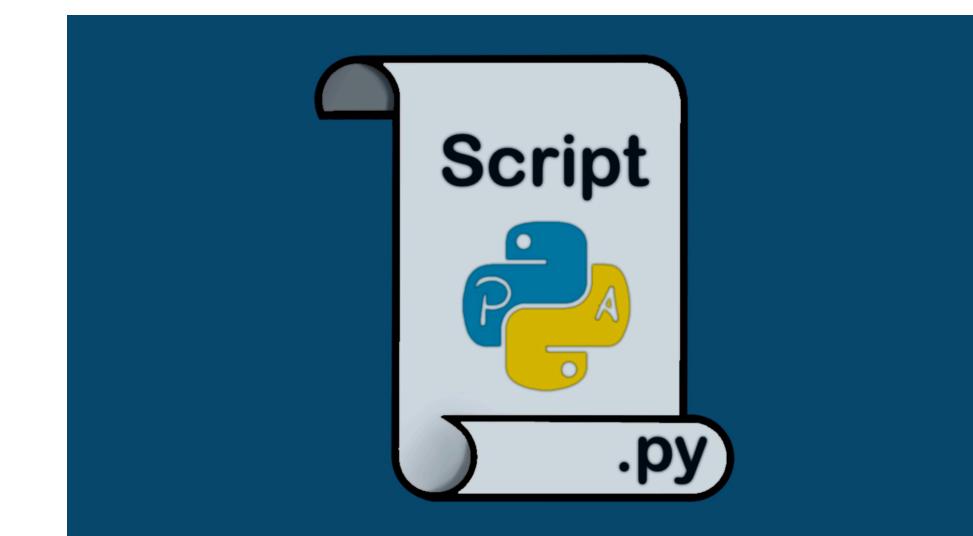


=



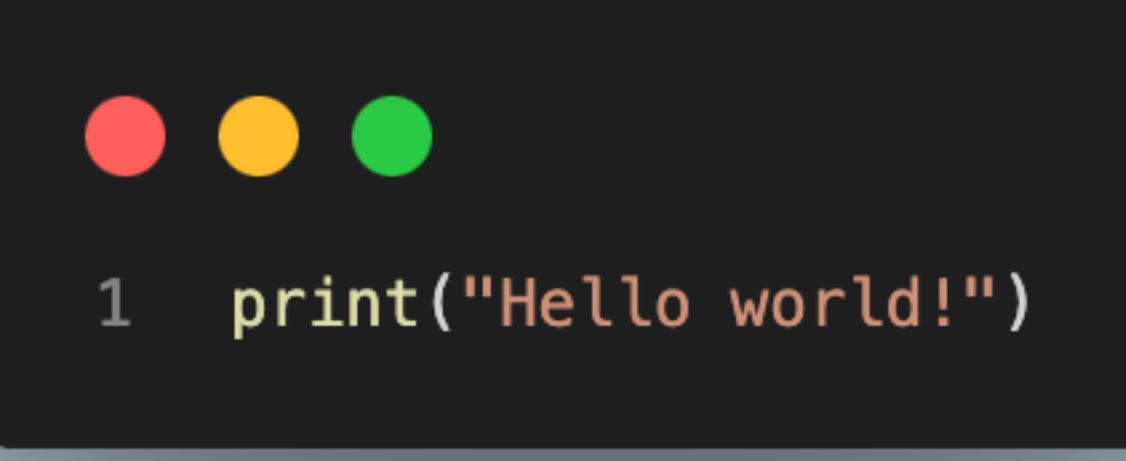
Почему Python?

Универсальность



Почему Python?

Лаконичный и понятный синтаксис



```
1 print("Hello world!")
```

VS



```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6     cout << "Hello, world!" << endl;
7     return 0;
8 }
```

Почему Python?

Низкий порог входа (до какой-то степени)

What is the easiest programming language for beginners?



Python. Due to its relatively straightforward syntax and emphasis on eliminating clutter, fast-growing Python is often seen as the easiest programming language to learn.

6 дек. 2023 г.



CareerFoundry

<https://careerfoundry.com> › blog › web-development › e...

[What Is the Easiest Programming Language to Learn in 2024?](#)

Почему Python?

Множество дополнений, устанавливающихся одной командой

requests 2.31.0

`pip install requests`

Released: May 22, 2023

Python HTTP for Humans.

Navigation

- Project description**
- Release history
- Download files

Project links

- Homepage
- Documentation
- Source

Statistics

Project description

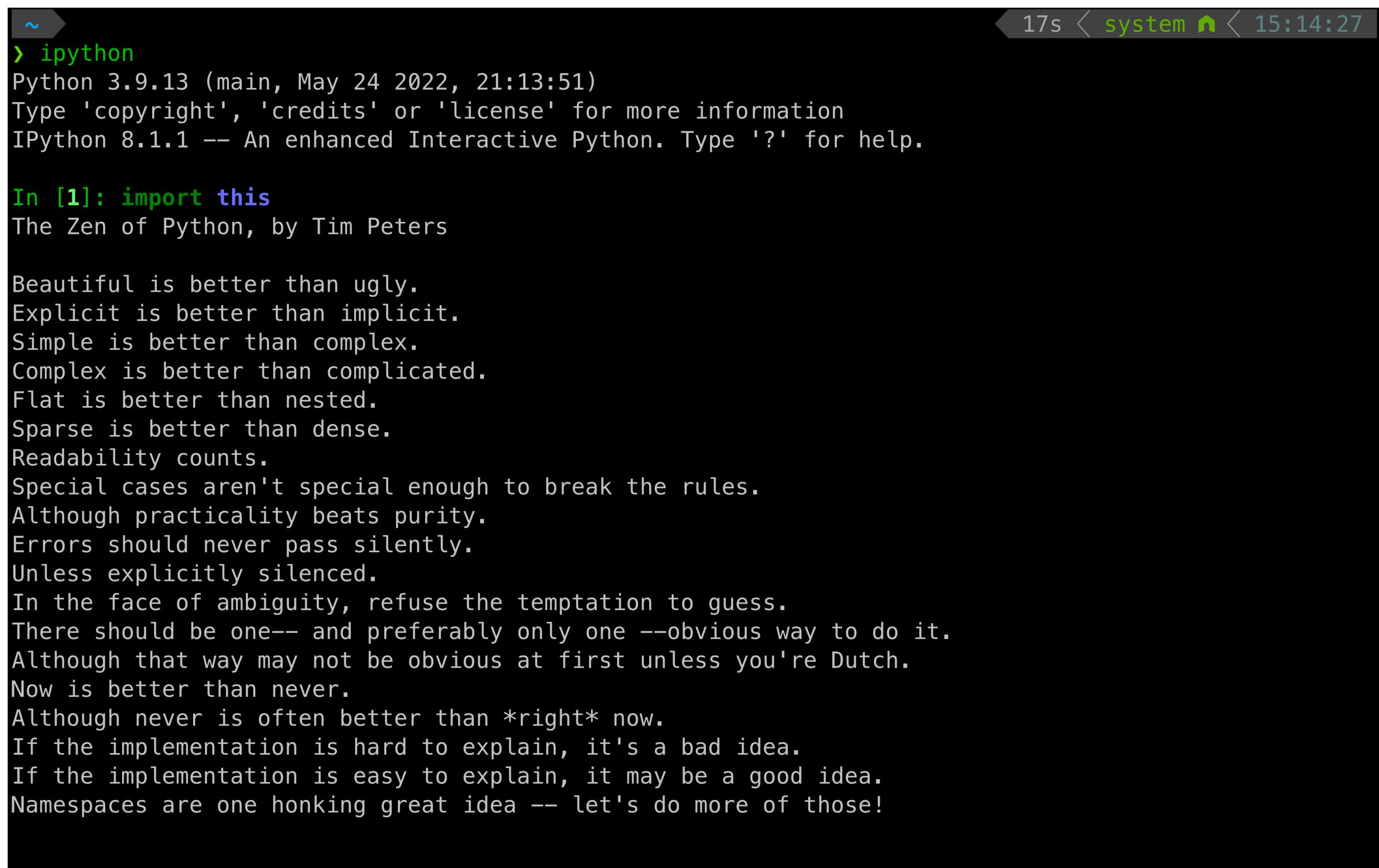
Requests

Requests is a simple, yet elegant, HTTP library.

```
>>> import requests
>>> r = requests.get('https://httpbin.org/basic-auth/user/pass', auth=('user', 'pass'))
>>> r.status_code
200
>>> r.headers['content-type']
'application/json; charset=utf8'
>>> r.encoding
'utf-8'
>>> r.text
'{"authenticated": true, ...'
>>> r.json()
{'authenticated': True, ...}
```

Почему Python?

Понятная философия



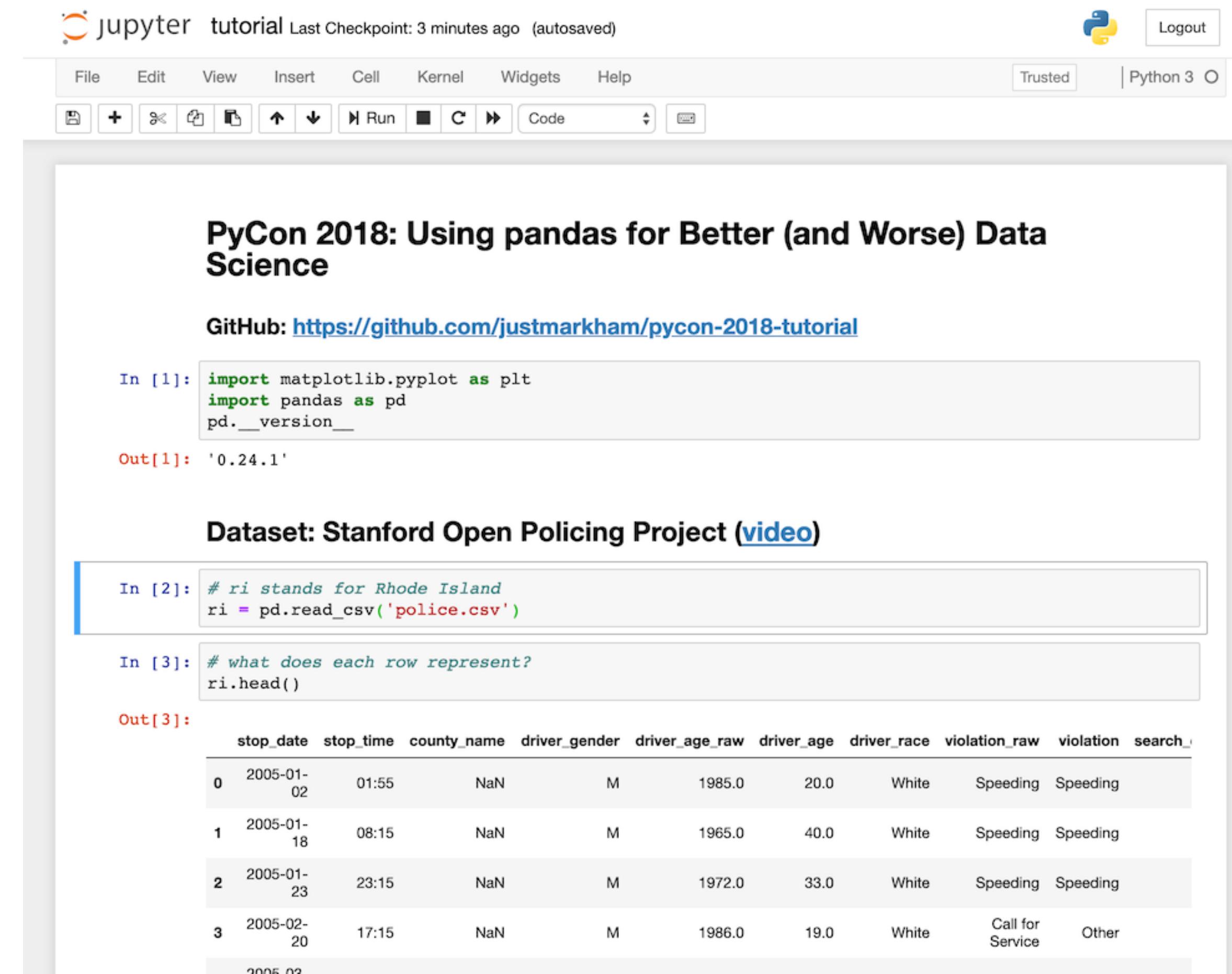
```
~> ipython
Python 3.9.13 (main, May 24 2022, 21:13:51)
Type 'copyright', 'credits' or 'license' for more information
IPython 8.1.1 -- An enhanced Interactive Python. Type '?' for help.

In [1]: import this
The Zen of Python, by Tim Peters

Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated.
Flat is better than nested.
Sparse is better than dense.
Readability counts.
Special cases aren't special enough to break the rules.
Although practicality beats purity.
Errors should never pass silently.
Unless explicitly silenced.
In the face of ambiguity, refuse the temptation to guess.
There should be one-- and preferably only one --obvious way to do it.
Although that way may not be obvious at first unless you're Dutch.
Now is better than never.
Although never is often better than *right* now.
If the implementation is hard to explain, it's a bad idea.
If the implementation is easy to explain, it may be a good idea.
Namespaces are one honking great idea -- let's do more of those!
```

Почему Python?

Интерпретируемость (это и плюс, и минус)



PyCon 2018: Using pandas for Better (and Worse) Data Science

GitHub: <https://github.com/justmarkham/pycon-2018-tutorial>

```
In [1]: import matplotlib.pyplot as plt
import pandas as pd
pd.__version__
```

```
Out[1]: '0.24.1'
```

Dataset: Stanford Open Policing Project ([video](#))

```
In [2]: # ri stands for Rhode Island
ri = pd.read_csv('police.csv')
```

```
In [3]: # what does each row represent?
ri.head()
```

```
Out[3]:
```

	stop_date	stop_time	county_name	driver_gender	driver_age_raw	driver_age	driver_race	violation_raw	violation	search_
0	2005-01-02	01:55	NaN	M	1985.0	20.0	White	Speeding	Speeding	
1	2005-01-18	08:15	NaN	M	1965.0	40.0	White	Speeding	Speeding	
2	2005-01-23	23:15	NaN	M	1972.0	33.0	White	Speeding	Speeding	
3	2005-02-20	17:15	NaN	M	1986.0	19.0	White	Call for Service	Other	

Где будем писать код?

IDE – Integrated Development Environment

Где будем писать код?

PyCharm

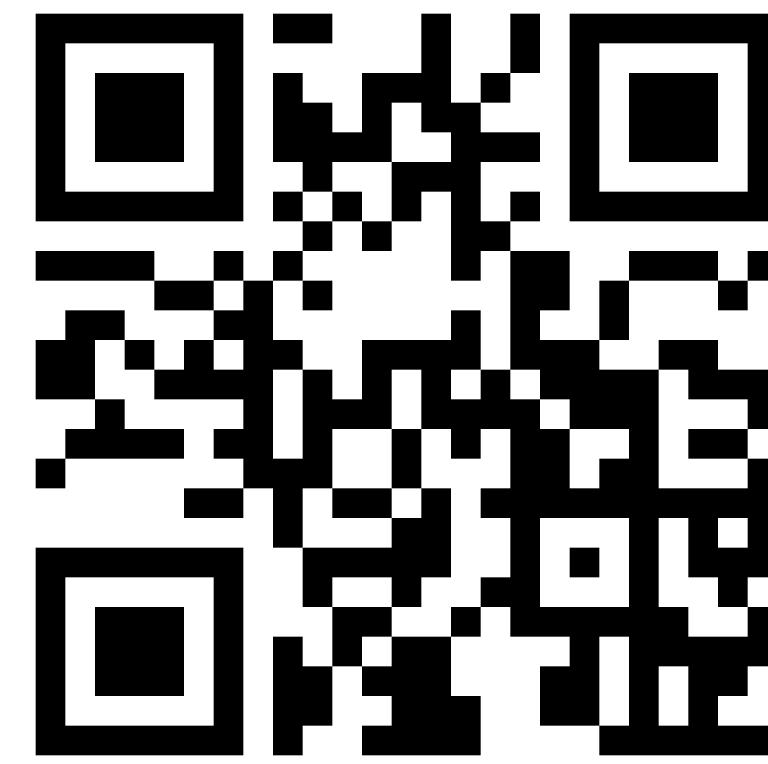
- + Множество удобных инструментов из коробки
- + Простая установка
- + Есть множество дополнений
- + Самый интеллектуальный из всех IDE для Python
- Немного сложноват поначалу



Где будем писать код?

Jupyter Notebook

- + Легко ставится, не надо ничего настраивать
- + Пошаговое выполнение кода
- + Можно писать комментарии текстом и разметкой, выводить картинки, графики и тд
- Из-за пошагового выполнения кода можно посадить кучу багов
- Не для всех задач удобен



Проще всего ставится через Anaconda

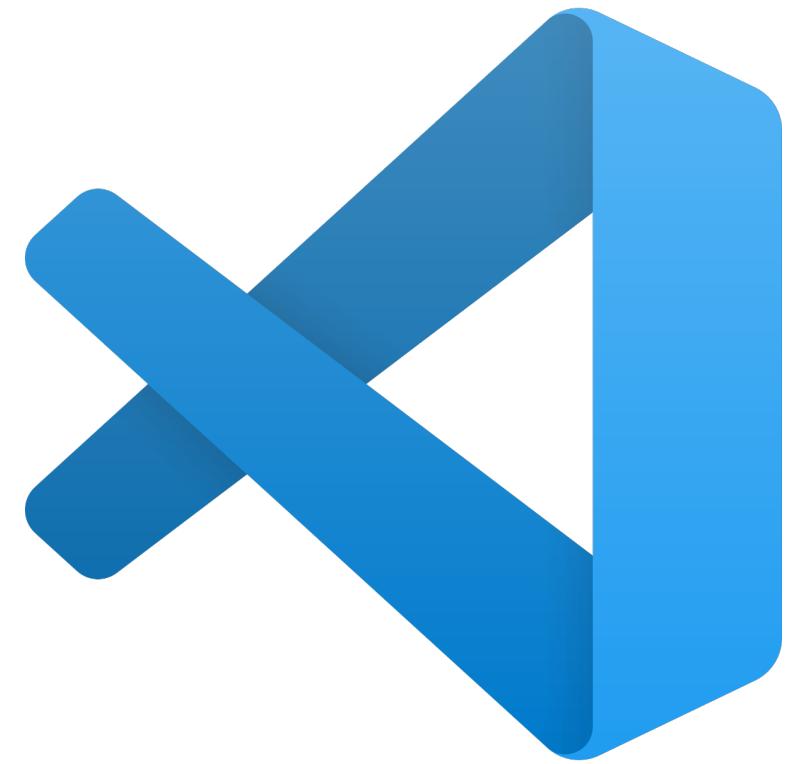
Еще вариант: colab.research.google.com



Где будем писать код?

VS Code

- + Лаконичный и красивый
- + Отлично работает на любой платформе
- + Дополнения на любой вкус и цвет
- + Дружит вообще с любыми языками программирования
- Может быть проблематично поженить его в питоном в некоторых случаях
- Ограниченные возможности в дебаге (поиске ошибок) относительно других IDE



А еще мы хотим сделать курс максимально крутым и ламповым
Поэтому заполните формочку

