



Python_lecture_final_final.pptx

04.05.2024

Что мы теперь умеем?

Программирование на уровне **Junior** и **Middle**



- Синтаксис Python (функции, классы, типы данных, файлы, стандартная библиотека)
- Работа с библиотеками и с ошибками
- IDE (Jupyter, PyCharm, VSCode)
- Виртуальные окружения
- Работа с ОС, взаимодействие с Linux
- Итераторы, генераторы, декораторы, контекстный менеджер

Важные концепции:

- Git, GitHub
- ООП
- Функциональное программирование
- Параллельное программирование
- Дебаггинг, профилирование
- Базы данных
- Интернет: устройство, общение, веб-разработка
- Регулярные выражения



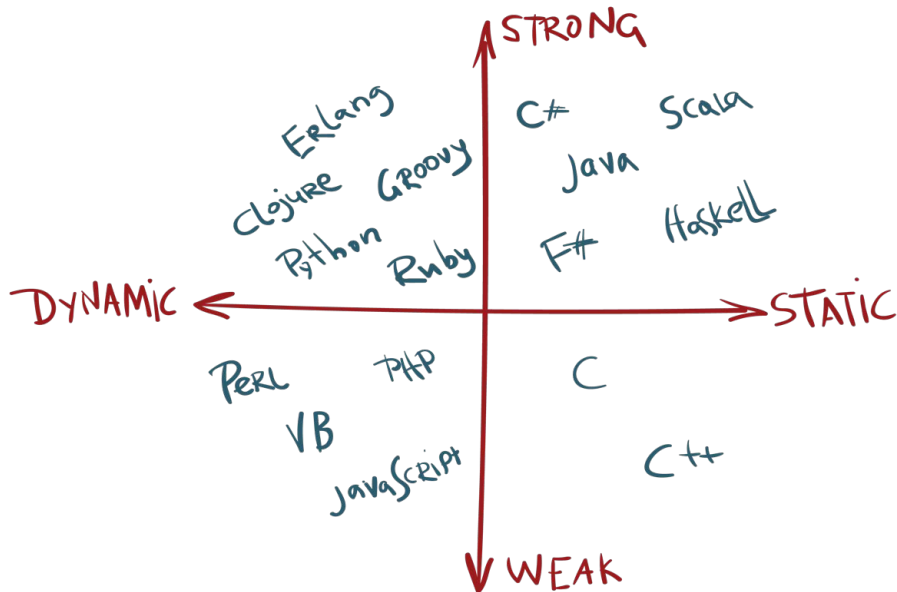
Анализ данных, биоинформатика

- Numpy, Pandas, работа с таблицами (analysis, filtering, merging, grouping)
- Визуализация, Matplotlib, Seaborn
- Специальные библиотеки (biopython, bioframe, ...)
- Статистика и ML в python



Насколько глубокий наш удав?

Python - какой язык?

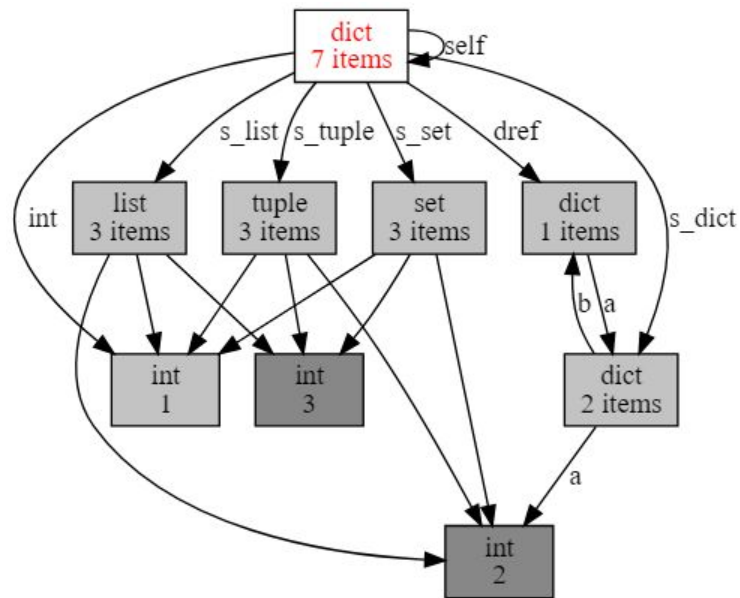




Насколько глубокий наш удав?

Python - какой язык?

Что такое **переменная**?



objgraph.readthedocs.io/en/stable/objgraph.html



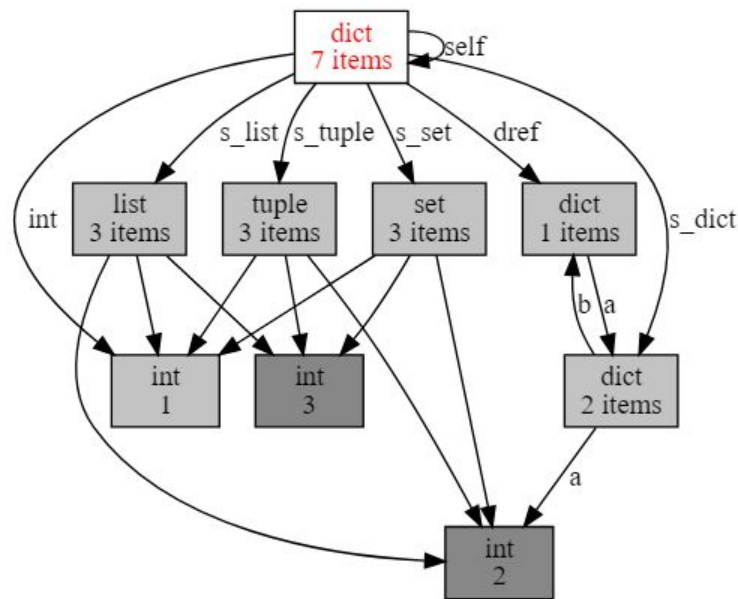
Насколько глубокий наш удав?

Python - какой язык?

Что такое **переменная**?

```

v struct _object {
    // ob_tid stores the thread id (or zero). It is also used by the GC and the
    // trashcan mechanism as a linked list pointer and by the GC to store the
    // computed "gc_refs" refcount.
    uintptr_t ob_tid;
    uint16_t _padding;
    struct _PyMutex ob_mutex; // per-object lock
    uint8_t ob_gc_bits; // gc-related state
    uint32_t ob_ref_local; // local reference count
    Py_ssize_t ob_ref_shared; // shared (atomic) reference count
    PyObject *ob_type;
};
#endif
    
```



objgraph.readthedocs.io/en/stable/objgraph.html



Насколько глубокий наш удав?

Python - какой язык?

Что такое переменная?

Как работает **сборщик мусора**?

Что такое утечка памяти?





Насколько глубокий наш удав?

Python - какой язык?

Что такое переменная?

Как работает сборщик мусора?

Что такое утечка памяти?

Что такое **функция**?

```
def keyword      name      parameter
  ↓              ↓          ↓
def celsius_to_fahr(temp) :
  return 9/5 * temp + 32
  ↑              ↖
return statement return value
```



Насколько глубокий наш удав?

Python - какой язык?

Что такое переменная?

Как работает сборщик мусора?

Что такое утечка памяти?

Что такое функция?

Что такое лямбда-функция?

```
def keyword      name      parameter
  ↓              ↓          ↓
def celsius_to_fahr(temp) :
  return 9/5 * temp + 32
  ↑              ↑
return statement return value
```




Насколько глубокий наш удав?

Python - какой язык?

Что такое переменная?

Как работает сборщик мусора?

Что такое утечка памяти?

Что такое функция?

Что такое лямбда-функция?

Что такое args и kwargs?

```
def keyword      name      parameter
  ↓              ↓          ↓
def celsius_to_fahr(temp) :
  return 9/5 * temp + 32
  ↑              ↑
return statement return value
```



Насколько глубокий наш удав?

Python - какой язык?

Что такое переменная?

Как работает сборщик мусора?

Что такое утечка памяти?

Что такое функция?

Что такое лямбда-функция?

Что такое args и kwargs?

В чем разница между **list** и **tuple**?





Насколько глубокий наш удав?

Python - какой язык?

Что такое переменная?

Как работает сборщик мусора?

Что такое утечка памяти?

Что такое функция?

Что такое лямбда-функция?

Что такое args и kwargs?

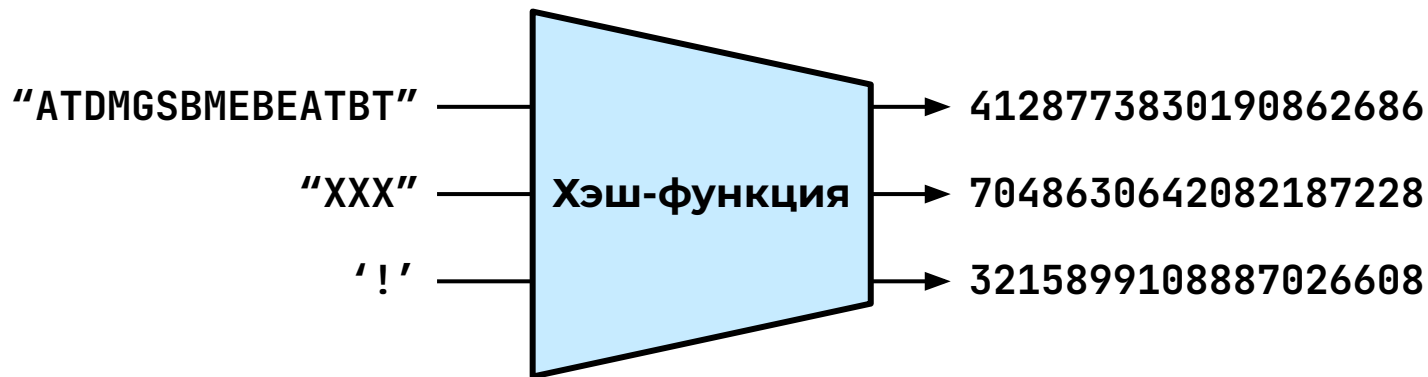
В чем разница между list и tuple?

Что может быть ключом в **словаре**?





Хэширование



По каждому входу X однозначно и (в идеале) легко вычисляется выход Y

По каждому выходу Y сложно или (в идеале) невозможно вычислить вход X

Минимальные изменения в X (в идеале) должны непредвиденно менять Y

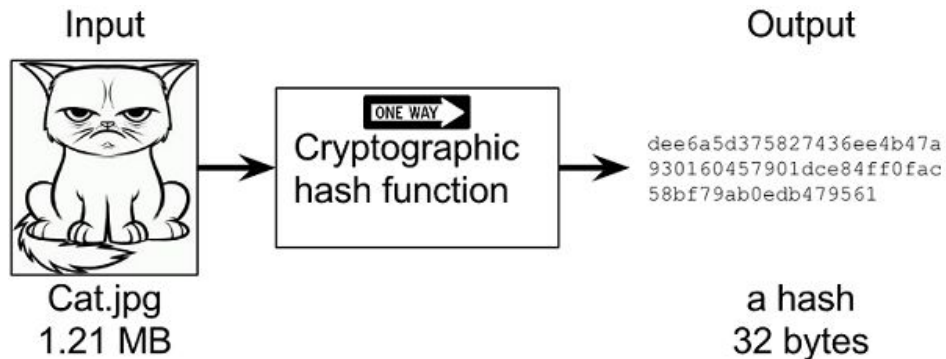
Независимо от типа и размера X , все результаты Y должны быть одним типом данных заданного размера

Разные входы X не должны приводить к одному выходу Y (“коллизия”)

Хэширование

Задачи:

- Хранение паролей
- Быстрый поиск
- Контрольные суммы
- Сканирование нежелательного контента
- ...



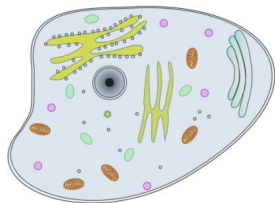


Насколько глубокий наш удав?

Что такое **ООП**?
Назовите 3 кита ООП.



Приходите к нам на python в ИБ, у нас есть...



Инкапсуляция



Полиморфизм

пока нет

Наследование



Насколько глубокий наш удав?

Что такое ООП?

Назовите 3 кита ООП.

Что такое **dunder**-методы?

Какие вы можете назвать?

Что делает ``__init__``?





Насколько глубокий наш удав?

Что такое ООП?

Назовите 3 кита ООП.

Что такое dunder-методы?


Какие вы можете назвать?

Что делает `__init__``?

Что такое **итератор**?


Что такое **генератор**?

Зачем нужны *iter* и *next*?



[파이썬] **iterable** 과 **iterator** 의 차이가 뭐예요?

헷갈리는 개념을 쉽게, 직관적으로, 하지만 깊게 공부하는 **뭐예요?** 시리즈





Насколько глубокий наш удав?

Что такое ООП?

Назовите 3 кита ООП.

Что такое dunder-методы?

Какие вы можете назвать?

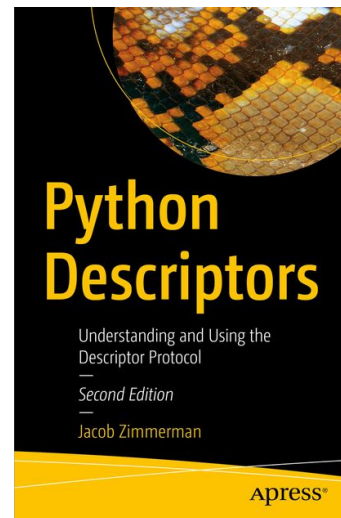
Что делает `__init__`?

Что такое итератор?

Что такое генератор?

Зачем нужны *iter* и *next*?

Что такое **дескрипторы**?





Насколько глубокий наш удав?

Что такое ООП?

Назовите 3 кита ООП.

Что такое dunder-методы?

Какие вы можете назвать?

Что делает `__init__`?

Что такое итератор?

Что такое генератор?

Зачем нужны `iter` и `next`?

Что такое дескрипторы?

Что такое **контекстный менеджер**?

Из чего состоит эффективный менеджер





Насколько глубокий наш удав?

Что такое **функциональное программирование**?

Что такое чистые функции?

Что такое функции высшего порядка?





Насколько глубокий наш удав?

Что такое функциональное программирование?

Что такое чистые функции?

Что такое функции высшего порядка?

Что такое **декоратор**?

Как можно написать декоратор?

Зачем нужен *functools.wraps*?

Какие декораторы вы помните?





Насколько глубокий наш удав?

Что такое функциональное программирование?

Что такое чистые функции?

Что такое функции высшего порядка?

Что такое декоратор?

Как можно написать декоратор?

Зачем нужен *functools.wraps*?

Какие декораторы вы помните?

Что такое **область видимости**?

Что такое **замыкание**?





Насколько глубокий наш удав?

В чем разница между **процессами** и **потоками**?

Ядро



Процесс



Поток



1 ядро



много ядер



2 процесса
на 1 ядро



5 процессов на
3 ядра



1 процесс обработался
в 1 поток на одном ядре



1 процесс сгенерировал много
потоков на одном ядре





Насколько глубокий наш удав?

В чем разница между процессами и потоками?

Что такое **IO**- и **CPU**-bound задани?

Ядро



Процесс



Поток



1 ядро



много ядер



2 процесса
на 1 ядро



5 процессов на
3 ядра



1 процесс обработался
в 1 поток на одном ядре



1 процесс сгенерировал много
потоков на одном ядре





Насколько глубокий наш удав?

В чем разница между процессами и потоками?

Что такое IO- и CPU-bound задани?

Что такое **GIL**?





Насколько глубокий наш удав?

В чем разница между **реляционными** и нереляционными **базами данных**?

SELECT * FROM



Select * From



select * from



SeLEct * fRoM

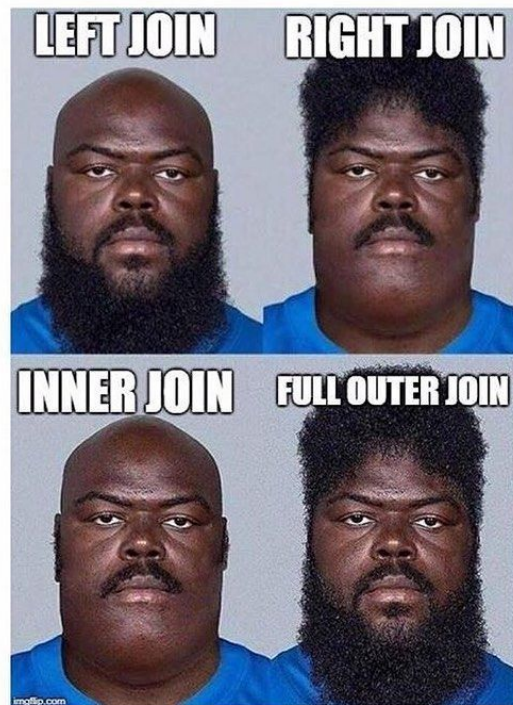




Насколько глубокий наш удав?

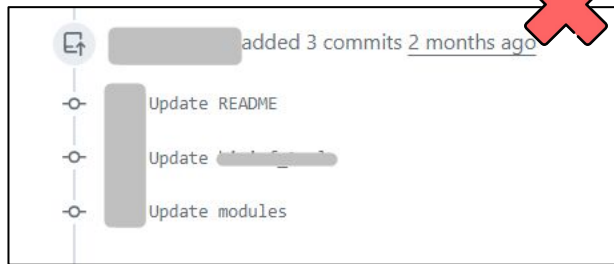
В чем разница между **реляционными** и нереляционными **базами данных**?

Какие типы **join** вы знаете?



Правила оформления

Какие вы можете назвать правила оформления кода?
Коммитов?
Порядок импортов?



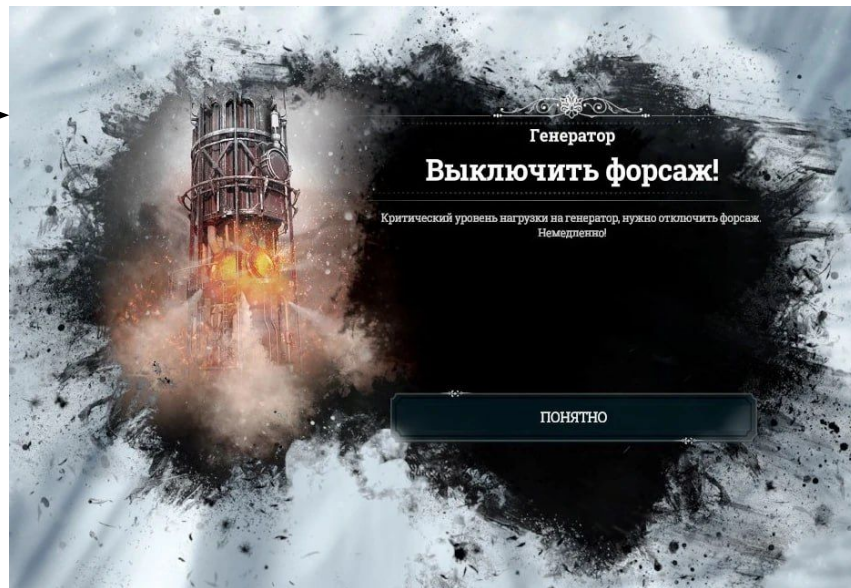


ИБ закончился: что делать дальше?

ИБ закончился: что делать дальше?

- Отдых

Вы пережили спринт
длинною в год



ИБ закончился: что делать дальше?

- Отдых

Отдых

Материал из Википедии — свободной энциклопедии

Текущая версия страницы пока не проверялась опытными участниками и может значительно отличаться от версии, проверенной 30 марта 2023 года. проверки требуют 7 правок.

У этого термина существуют и другие значения, см. *Отдых (значения)*.

Отдых — состояние покоя, либо времяпрепровождение, целью которого является восстановление сил, достижение работоспособного состояния организма (рекреация)^[1]; время, свободное от работы^[2]. Вынужденное бездействие — такое, как ожидание или отбывание наказания, болезненное состояние, — отдыхом не является.

Обычно отдых включает в себя следующие стадии: усталость, расслабление, восстановление сил, развлечение. Занятия для отдыха в стадии развлечения: игры (в том числе компьютерные), приём пищи, беседа, любительский спорт, рекреационное рыболовство, прослушивание музыки, радио, просмотр телевизора, произведений искусства, занятия различными видами искусств, чтение книг, пикник, поход в баню (сауну), волонтерство и другие развлечения и виды деятельности.

Содержание [скрыть]

1 История

2 Классификация

2.1 По продолжительности

2.2 По активности



Полезные ресурсы:

- [Aviasales](#)
- [Kinopoisk](#)
- [Steam](#)





ИБ закончился: что делать дальше?

- Отдых
- Саморефлексия



Вы пережили спринт
длинною в год



- Оглянитесь, посмотрите чего вы добились?
- Запишите куда-нибудь свои мысли, чувства
А тем более сохраните мемы из чатов
- **Вспомните а зачем вы поступали в ИБ?**
Откройте и перечитайте свою мотивашку
Сравните с тем что в реальности.
Что удалось? Что пере-удалось? Что недо-
удалось?
- А в чем была глобальная цель?





ИБ закончился: что делать дальше?

- Отдых
- Саморефлексия
- Отдохнул, отрефлексировал: что учить дальше?

ИБ закончился: что делать дальше?

- Отдых
- Саморефлексия
- Отдохнул, отрефлексировал: что учить дальше?

**ХВАТИТ ЗАНИМАТЬСЯ ЕРУНДОЙ В ИНТЕРНЕТЕ!
ЗАРПЛАТА ЗАВИСИТ ОТ РАБОТЫ. РАБОТАЙ!**



ИБ закончился: что делать дальше?

- Отдых
- Саморефлексия
- Работаем

Институт биоинформатики сильно развращает:

- Сложные, но ясные и четкие задания
- Понятные дедлайны
- Правильные ответы, поддерживающий фидбек
- Комьюнити

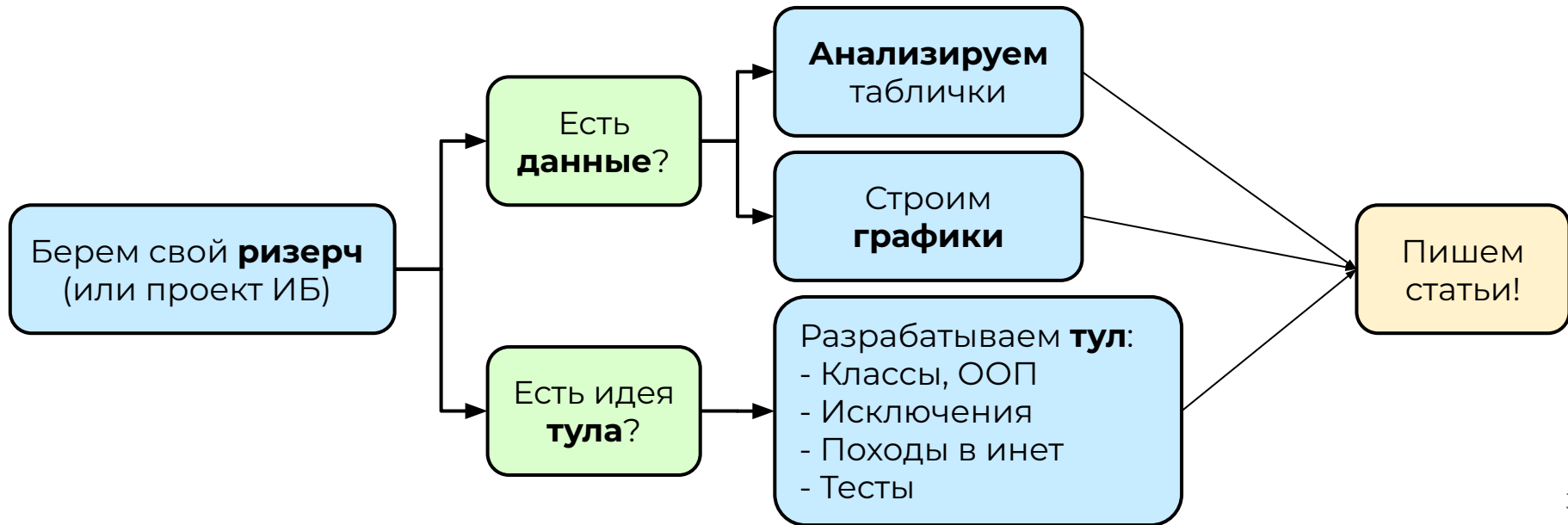
В обычной работе такого будет очень мало.
Хотя все зависит от вас.





ИБ закончился: что делать дальше?

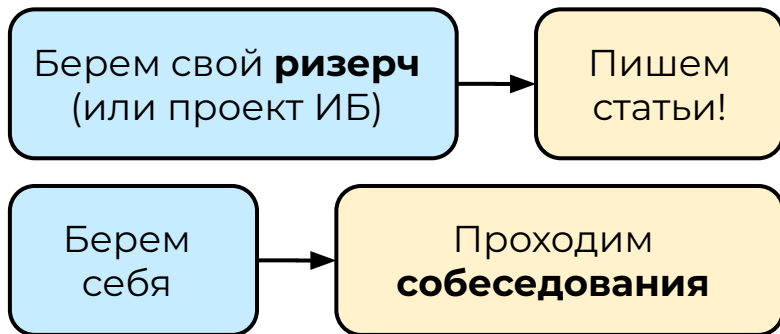
- Отдых
- Саморефлексия
- Работаем





ИБ закончился: что делать дальше?

- Отдых
- Саморефлексия
- Работаем

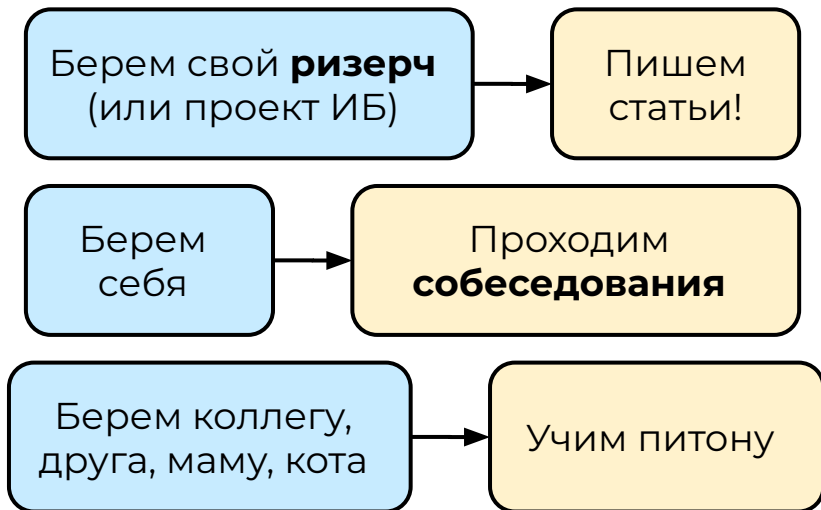


Даже если вам прям сейчас не надо: лучше иметь опыт
Собеседования: тоже навык
Не начинайте собеседования с компании мечты:)



ИБ закончился: что делать дальше?

- Отдых
- Саморефлексия
- Работаем





ИБ закончился: что делать дальше?

- Отдых
- Саморефлексия
- Работаем
- Ну прям очень хочется что-то поизучать...



Куда дальше учиться в проге?

• Python:

ООП, ФП, Параллельное и асинхронное программирование, кишки и устройство питона, компилирование и ускорение

[Курсы CSC](#)

[Прикладной Python. VK](#)

[Никита Соболев: курс про питон с кишками](#)

[AGLab: Coding our own FastQC from scratch](#)

[Сколько строк на C нужно, чтобы выполнить \$a + b\$ в Python?](#)

Доклады Никиты Соболева, Златы Обуховской, David beazley

• Git:

Исправление ошибок, rebase, cherry-pick, решение конфликтов

[Advanced Git Tutorial](#) от [freeCodeCamp.org](#)

[So You Think You Know Git - FOSDEM 2024](#)

[Pro Git book](#)

• ОС, Линукс

[Что такое операционная система и как она работает](#)

[Архитектура операционной системы. CSC](#)

[Основы программирования для Linux](#)

[Системные вызовы в операционной системе](#)

• Алгоритмы

[Т. Хирьянов, Алгоритмы и структуры данных на Python 3](#)

[Курсы CSC](#)

[LeetCode](#), [CodeWars](#), [Rosalind](#)

[Тренировки по алгоритмам от Яндекса](#)

• Вокруг проги:

Докер: [курс Антона](#)

CI/CD ([Habr](#), [GitHub](#)), [GitHub Actions](#) ([GitHub](#), [Tutorial](#))

Kubernetes ([Habr](#), [Tutorial](#))

SQL: [SQL от Karpov.Courses](#), [SQL от BEEGEEK](#) (Поколение Python)

[Karpov.Courses](#) - курсы для аналитиков данных

[Яндекс.Практикум](#)

• Блогеры

[Все доклады Григория Петрова на MoscowPython](#)

[Sergey Nemchinskiy](#)

[Хитрый питон](#)

[Moscow Python](#) (конференции и подкаст)

[Диджитализируй](#)

[Python Russian](#)

[Python Clinic](#)





На этом всё:)



Заданий пока нет
Заглядывайте позже.