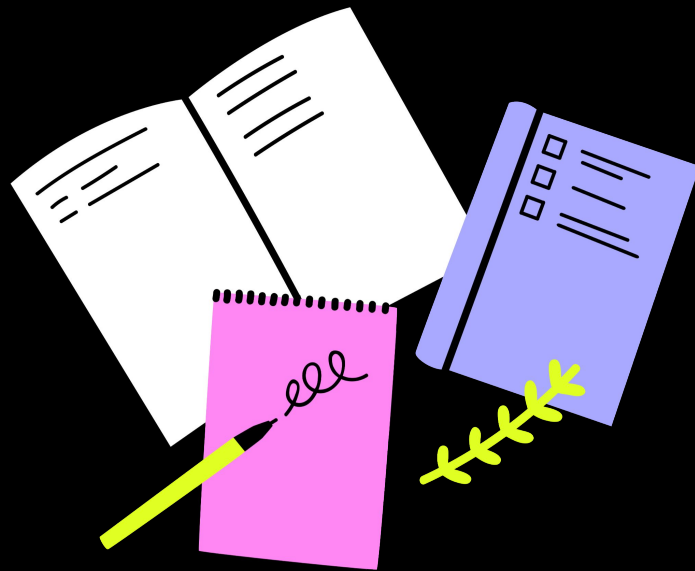




Семинар 1

Оформление ноутбука и закрепление функций и генераторов





Знакомство





Юлия Пономарева

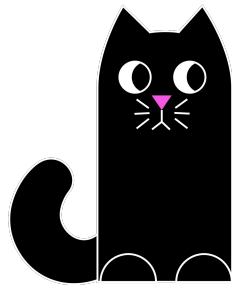
Machine Learning Engineer

Дарю зрение машинам и повышаю выручку компаний.

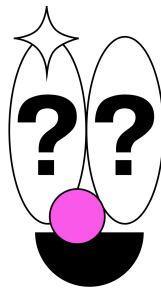
- 💡 Система распознавания товаров и ценников по фотографии
- 💡 Предсказание выгодной локации для торговой точки
- 💡 Анализ факторов на выручку

Теперь ваша очередь!

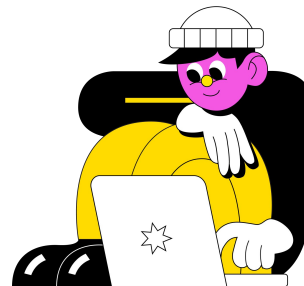
Ответьте на несколько вопросов сообщением в чат



Из какого вы города?



Сколько вам лет?



Кем вы работаете сейчас? Как долго?



Цели семинара №12:

- 📌 Узнать, как работать с jupyter notebook эффективней
- 📌 Научиться красиво и доступно оформлять jupyter notebook
- 📌 Посмотреть на функционал модуля Random
- 📌 Более детально изучить словари, функции в Python
- 📌 Разобраться с генераторами
- 📌 Познакомиться с list, set, dict comprehensions





Викторина

Минутка самопроверки



С помощью какого тега можно добавить изображение в markdown?

1. ``
2. `<image>`
3. `<picture>`
4. `<photo>`

<<0:30->>

С помощью какого тега можно добавить изображение в markdown?

1. ``
2. `<image>`
3. `<picture>`
4. `<photo>`



Какой будет вывод у этого кода?

1. ['a', 'b', 'c', 'd']
2. TypeError: say_hello() missing 1 required argument: 'a'
3. 'abcd'

```
def copy(a: list) -> list:  
    return a[:]  
  
copy('abcd')
```

<<0:30->>

Какой будет вывод у этого кода?

1. ['a', 'b', 'c', 'd']
2. TypeError: say_hello() missing 1 required argument: 'a'
3. 'abcd'

```
def copy(a: list) -> list:  
    return a[:]
```

```
copy('abcd')
```

'abcd'



Что вернет данный код?

1. 7000
2. None
3. 8000
4. Key Error

```
data = {  
    'name': 'John',  
    'age': 20,  
    'salary': 8000,  
    'city': 'Boston'  
}  
  
data.setdefault('salary', 7000)
```

<<0:30->>

Что вернет данный код?

1. 7000
2. None
3. 8000
4. Key Error

```
data = {  
    'name': 'John',  
    'age': 20,  
    'salary': 8000,  
    'city': 'Boston'  
}  
  
data.setdefault('salary', 7000)
```

8000



С помощью какой функции из модуля Random можно получить случайные объекты из списка?

1. `random.sample()`
2. `random.randint()`
3. `random.random()`
4. `random.choice()`

<<0:30->>

С помощью какой функции из модуля Random можно получить случайные объекты из списка?

1. `random.sample()`
2. `random.randint()`
3. `random.random()`
4. `random.choice()`

```
a = ['red', 'yellow', 'black', 'red']  
random.choice(a)  
  
'black'
```

```
random.sample(a, k=1)  
  
['yellow']
```



Какой будет вывод у этого кода?

```
def say_hello(name, greeting='Hello'):  
    return f'{greeting}, {name}'
```

```
say_hello(greeting='Hi', 'John')
```

1. Hi, John
2. SyntaxError: positional argument follows keyword argument
3. TypeError: say_hello() missing 1 required positional argument: 'name'
4. Hello, John

<<0:30->>

Какой будет вывод у этого кода?

```
def say_hello(name, greeting='Hello'):  
    return f'{greeting}, {name}'
```

```
say_hello(greeting='Hi', 'John')
```

```
File "<ipython-input-13-66cbe01dd924>", line 1
```

```
    say_hello(greeting='Hi', 'John')
```

^

SyntaxError: positional argument follows keyword argument

1. Hi, John
2. **SyntaxError: positional argument follows keyword argument**
3. TypeError: say_hello() missing 1 required positional argument: 'name'
4. Hello, John



Выберите особенности генератора в Python

1. Может обрабатывать большие объемы данных без переполнения памяти
2. Хранит в памяти все элементы
3. Хранит в памяти последний вычисленный элемент
4. Может обрабатывать только небольшие наборы данных

<<0:30->>

Выберите особенности генератора в Python

1. Может обрабатывать большие объемы данных без переполнения памяти
2. Хранит в памяти все элементы
3. Хранит в памяти последний вычисленный элемент
4. Может обрабатывать только небольшие наборы данных



Какой будет вывод у этого кода?

```
[i if i % 2 == 0 for i in range(5)]
```

1. SyntaxError: invalid syntax
2. [0, 2, 4]
3. [2, 4]
4. [1, 3]

<<0:30->>

Какой будет вывод у этого кода?

```
[i if i % 2 == 0 for i in range(5)]
```

```
File "<ipython-input-21-ef3cb1223bea>", line 1
```

```
[i if i % 2 == 0 for i in range(5)]
```

^

```
SyntaxError: invalid syntax
```

1. `SyntaxError: invalid syntax`
2. `[0, 2, 4]`
3. `[2, 4]`
4. `[1, 3]`



С помощью фигурных скобок {} какие создаются comprehensions?

1. set comprehensions
2. dict comprehensions
3. tuple comprehensions
4. list comprehensions

<<0:30->>

С помощью фигурных скобочек {} какие создаются comprehensions?

1. set comprehensions
2. dict comprehensions
3. tuple comprehensions
4. list comprehensions

```
{i for i in range(5)}
```

```
{0, 1, 2, 3, 4}
```

```
{i: i for i in range(5)}
```

```
{0: 0, 1: 1, 2: 2, 3: 3, 4: 4}
```



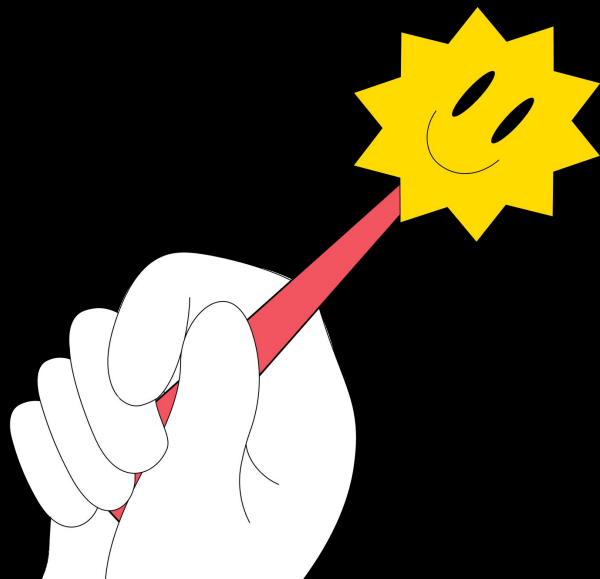
Ваши вопросы?





Практика

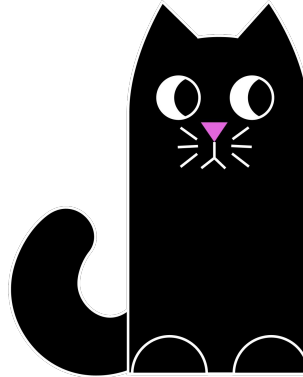
Оформление ноутбука и закрепление функций и генераторов



Задание 0.

Весь ноутбук оформляйте с заголовками и текстами:

- ☀ Номер задачи - заголовок 2
- ☀ Номер подзадачи - заголовок 3
- ☀ Предоставленные наборы данных оформляйте, как код



Вся
практика



Задание 1.

1.1 Соедините два словаря в один

```
dict1 = {'One': 1, 'Two': 2, 'Three': 3}  
dict2 = {'Four': 4, 'Five': 5, 'Six': 6}
```

1.2 Напишите функцию, которая на вход принимает два словаря и возвращает один объединенный словарь

Используйте аннотирование типов



5 минут



Задание 1.

1.1 Соедините два словаря в один

```
dict1 = {'One': 1, 'Two': 2, 'Three': 3}  
dict2 = {'Four': 4, 'Five': 5, 'Six': 6}
```

1.2 Напишите функцию, которая на вход принимает два словаря и возвращает один объединенный словарь

Используйте аннотирование типов



Задание 2.

Напишите функцию, которая из двух списков, делает один словарь, где элементы из первого списка - ключи, а элементы из второго списка - значения

```
keys = ['One', 'Two', 'Three']  
values = [1, 2, 3]
```

Используйте аннотирование типов

2.1 Используя цикл for

2.2 Используя dict comprehensions



5 минут



Задание 2.

Напишите функцию, которая из двух списков, делает один словарь, где элементы из первого списка - ключи, а элементы из второго списка - значения

```
keys = ['One', 'Two', 'Three']  
values = [1, 2, 3]
```

Используйте аннотирование типов

2.1 Используя цикл for

2.2 Используя dict comprehensions



Задание 3.

Извлеките только два ключа name и age из представленного словаря

3.1 Напишите функцию с циклом for

Функция на вход принимает:

- ☀ исходный словарь
- ☀ ключи, которые нужно извлечь (аргумент по умолчанию)

На выходе словарь с нужными ключами

Используйте аннотирование типов

3.2 Используя dict comprehensions

```
client_dict = {  
    "name": "John",  
    "age": 25,  
    "salary": 5000,  
    "city": "Moscow"  
}
```

5 минут



Задание 3.

Извлеките только два ключа name и age из представленного словаря

3.1 Напишите функцию с циклом for

Функция на вход принимает:

- ☀ исходный словарь
- ☀ ключи, которые нужно извлечь (аргумент по умолчанию)

На выходе словарь с нужными ключами

Используйте аннотирование типов

3.2 Используя dict comprehensions

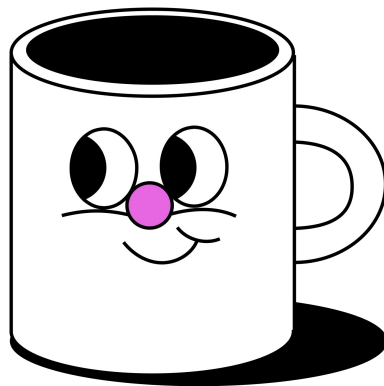
```
client_dict = {  
    "name": "John",  
    "age": 25,  
    "salary": 5000,  
    "city": "Moscow"  
}
```



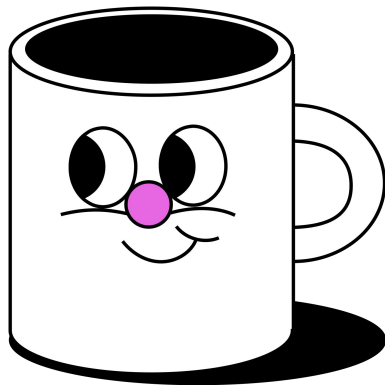
<<5:00



Перерыв



Перерыв



<<5:00->>

Задание 4.

4.1 Сгенерируйте случайные целые числа от 0 до 100 в количестве 5 штук с помощью модуля random

- Зафиксируйте псевдогенерацию, чтобы сгенерированные значения всегда были одинаковые
- Используйте list comprehensions



5 минут



Задание 4.

4.1 Сгенерируйте случайные целые числа от 0 до 100 в количестве 5 штук с помощью модуля random

- Зафиксируйте псевдогенерацию, чтобы сгенерированные значения всегда были одинаковые
- Используйте list comprehensions



4.2 Напишите генератор

- ☀ Генератор на вход принимает список с данными о клиенте (данные из пункта 4.1)
- ☀ Внутри генератора реализуйте обход по списку с данными

На каждой итерации генератор будет возвращать кортеж из двух элементов:

1. данные по клиенту (в зависимости от итерации, на 0 итерации вернется 0 элемент, на 1 итерации вернется 1 элемент и тд)
2. целочисленное значение, которое показывает, сколько секунд прошло с предыдущей итерации

Примечание: секунды, которые возвращаются должны показывать время не с начала запуска генератора, а именно то время, которое прошло с предыдущей итерации. А значит время на первой итерации должно равняться 0.

- ☀ Используйте функцию `time` из модуля `time` для подсчета времени.
- ☀ Чтобы проверить работу таймера, запустите проход по генератору в цикле с `time.sleep(2)`



10 минут



4.2 Напишите генератор

- ☀ Генератор на вход принимает список с данными о клиенте (данные из пункта 4.1)
- ☀ Внутри генератора реализуйте обход по списку с данными

На каждой итерации генератор будет возвращать кортеж из двух элементов:

1. данные по клиенту (в зависимости от итерации, на 0 итерации вернется 0 элемент, на 1 итерации вернется 1 элемент и тд)
2. целочисленное значение, которое показывает, сколько секунд прошло с предыдущей итерации

Примечание: секунды, которые возвращаются должны показывать время не с начала запуска генератора, а именно то время, которое прошло с предыдущей итерации. А значит время на первой итерации должно равняться 0.

- ☀ Используйте функцию `time` из модуля `time` для подсчета времени.
- ☀ Чтобы проверить работу таймера, запустите проход по генератору в цикле с `time.sleep(2)`



Задание 5.

5.1 Найдите картинку в интернете и прикрепите её в ячейку с текстом

5.2 Создайте следующую таблицу в ячейке с текстом

	ДО СОБЫТИЯ	ВО ВРЕМЯ СОБЫТИЯ
1 контрольная точка	250	500
2 контрольная точка	350	700
3 контрольная точка	400	800
Итого за победу	1 000	2 000

5 минут



Задание 5.

5.1 Найдите картинку в интернете и прикрепите её в ячейку с текстом

5.2 Создайте следующую таблицу в ячейке с текстом

	ДО СОБЫТИЯ	ВО ВРЕМЯ СОБЫТИЯ
1 контрольная точка	250	500
2 контрольная точка	350	700
3 контрольная точка	400	800
Итого за победу	1 000	2 000



Задание 6.

Напишите функцию, которая может принимать любое количество трат пользователя и считать сумму и среднее.

- ☀ На вход поступают целочисленные значения в любом количестве
- ☀ На выходе словарь с ключами суммы трат и средней траты

5 минут



Задание 6.

Напишите функцию, которая может принимать любое количество трат пользователя и считать сумму и среднее.

- ☀ На вход поступают целочисленные значения в любом количестве
- ☀ На выходе словарь с ключами суммы трат и средней траты



Ваши вопросы?

Подведем итоги





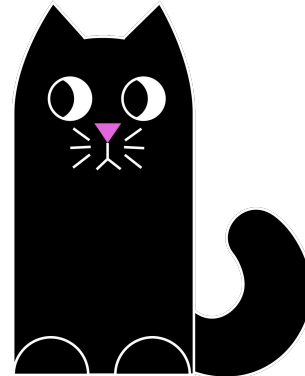
Домашнее задание



Домашнее задание 1

Оформляйте ноутбук, используя эти советы:

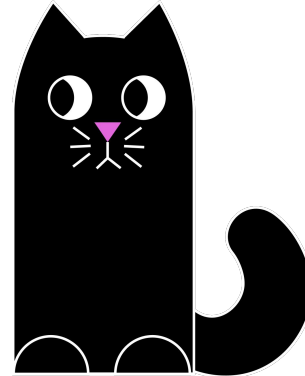
- ☀ Номер задачи - заголовок 2
- ☀ Номер подзадачи - заголовок 3
- ☀ Предоставленные наборы данных оформляйте, как код



Домашнее задание 2

На складе лежат разные фрукты в разном количестве.

Нужно написать функцию, которая на вход принимает любое количество названий фруктов и их количество, а возвращает общее количество фруктов на складе

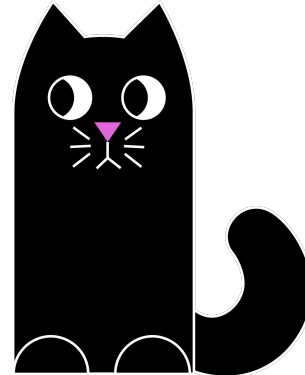


Домашнее задание 3

Дан список с затратами на рекламу. Но в данных есть ошибки, некоторые затраты имеют отрицательную величину. Удалите такие значения из списка и посчитайте суммарные затраты

[100, 125, -90, 345, 655, -1, 0, 200]

Используйте list comprehensions



Домашнее задание 4

Даны два списка.

Дата покупки

```
['2021-09-14', '2021-12-15', '2021-09-08', '2021-12-05', '2021-10-09', '2021-09-30', '2021-12-22', '2021-11-29', '2021-12-24', '2021-11-26', '2021-10-27', '2021-12-18', '2021-11-09', '2021-11-23', '2021-09-27', '2021-10-02', '2021-12-27', '2021-09-20', '2021-12-13', '2021-11-01', '2021-11-09', '2021-12-06', '2021-12-08', '2021-10-09', '2021-10-31', '2021-09-30', '2021-11-09', '2021-12-13', '2021-10-26', '2021-12-09']
```

Суммы покупок по датам

```
[1270, 8413, 9028, 3703, 5739, 4095, 295, 4944, 5723, 3701, 4471, 651, 7037, 4274, 6275, 4988, 6930, 2971, 6592, 2004, 2822, 519, 3406, 2732, 5015, 2008, 316, 6333, 5700, 2887]
```

4.1 Найдите, какая выручка у компании в ноябре

Используйте list comprehensions

4.2 Найдите выручку компании в зависимости от месяца

Для этого напишите функцию, которая на вход принимает список с датами и список с выручкой, а на выходе словарь, где ключи - это месяцы, а значения - это выручка.

Используйте аннотирование типов.





Спасибо
за внимание

A yellow smiley face is drawn over the text. It has two vertical lines for eyes and a curved line for a mouth, positioned to the right of the word 'Спасибо' and below the word 'за'.