

# Продолжение знакомства с jupyter notebook

Урок 1





# Знакомство и содержание урока



## 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
- ✦ Веду канал на youtube [machine learrning](#)



## Что будет на уроке сегодня

- 📌 Эффективная работа в jupyter notebook
- 📌 Модуль Random
- 📌 Детали Dict и функций
- 📌 Генераторы
- 📌 list, set, dict comprehensions





# Jupyter notebook



# Ячейки в jupyter notebook

В ноутбуках есть два вида ячеек:

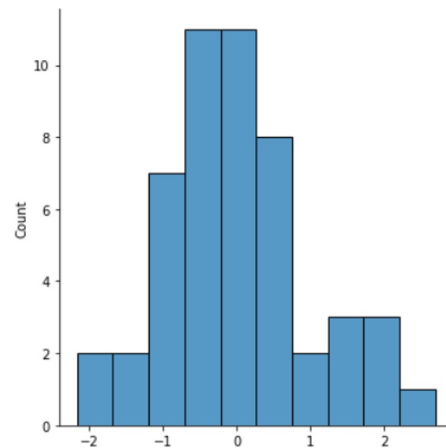
- с кодом
- с текстом

На графике ниже видим плотность распределения. Безумно красиво.

```
In [15]: import seaborn as sns
import numpy as np

x = np.random.randn(50)

sns.displot(x, bins=10);
```





## Ячейки в jupyter notebook

В ноутбуках есть два режима:

- режим редактирования
- режим ввода команд

```
In [ ]: x = np.random.randn(42)
```

```
In [ ]: x = np.random.randn(42)|
```

## Сочетания клавиш

Горячие сочетание клавиш (шорткат) — это нажатие кнопки/клавиши (или сочетания клавиш) на клавиатуре, которому назначено некое действие — команды, исполняемые данной системой.

Действие	Сочетание клавиш
Ctrl + A	Выделить все
Ctrl + Home	Подняться на начало ячейки
Ctrl + End	Опуститьсь на конец
Ctrl + Left	Перемещаться на одно слово влево
Ctrl + Right	Перемещаться на одно слово вправо

Действие	Сочетание клавиш
Tab	Автодополнение
Shift + Tab	Документация
Ctrl + Z	Отменить действие
Ctrl + /	Закомментировать или раскомментировать
Ctrl + D	Удалить целую строку



## Сочетания клавиш

Горячие сочетание клавиш (шорткат) — это нажатие кнопки/клавиши (или сочетания клавиш) на клавиатуре, которому назначено некое действие — команды, исполняемые данной системой.

Действие	Сочетание клавиш
A	Создать новую ячейку сверху
B	Создать новую ячейку снизу
X	Вырезать ячейку
C	Скопировать ячейку
V	Вставить ячейку

Действие	Сочетание клавиш
F	Найти и заменить
Y	Перевести ячейку в код
M	Перевести ячейку в текст
L	Показать нумерацию строк
H	Показать сочетания клавиш



# Markdown

**Markdown** - облегчённый язык разметки, созданный с целью обозначения форматирования в простом тексте, с максимальным сохранением его читаемости человеком, и пригодный для машинного преобразования в языки для продвинутых публикаций.

Этот язык разметки нужен для того, чтобы создавать красиво оформленные тексты. Вам не нужно использовать Microsoft Word или подобные программы, чтобы создавать документы с жирным или курсивным начертанием, цитатами, ссылками и таблицами.

```
# **жирный** |  
__fdgdfg__<br>  
*курсив*  
_fgfd_  
- список  
- список  
  
Latex  

$$\frac{1}{n} \sum x_{ij}$$
  
<img src=' '>
```

# Markdown

Для оформления и выделения мы используем символы, которые должны располагаться перед и после фразы или слова, которое нужно оформить.

Действие	Сочетание клавиш
<b>**</b> или <b>__</b>	Выделение жирным
<i>*</i> или <i>_</i>	Выделение курсивом
-	Создание списка
>	Создание цитаты



# Словарь



## Словарь

Словарь - неупорядоченная структура данных, которая позволяет хранить пары «ключ — значение».

```
client_data = {  
    'name': 'John',  
    'surname': 'Doe',  
    'age': 32  
}
```

```
client_data
```

```
{'name': 'John', 'surname': 'Doe', 'age': 32}
```



## Словарь

Вложенный словарь – это словарь, содержащий другие словари.

```
client_data.update({  
    'age': 15,  
    'body': {  
        'weight': 80,  
        'height': 186  
    }  
})
```

```
client_data
```

```
{'name': 'John',  
 'surname': 'Doe',  
 'age': 15,  
 'body': {'weight': 80, 'height': 186}}
```



# Модуль Random



## Модуль random

Модуль random предоставляет функции для генерации случайных чисел или к примеру, для случайного выбора элементов последовательности.

При разработке программ довольно часто возникает необходимость получить некоторое случайно выбранное из множества значение, к примеру, чтобы только на этом объекте показать проведенную работу аналитика, а не чтобы выводить весь массив данных.

```
random.randint(0, 10)
```

9

```
a = ['black', 'red', 'yellow', 'green']  
random.sample(a, 2)
```

```
['yellow', 'red']
```





# Аргументы функций



# Аргументы функций

**Позиционный аргумент** - это аргумент, передаваемый в функцию в определенной последовательности (на определенных позициях), без указания их имен

**Именованный аргумент** - это аргумент, передаваемые в функцию при помощи имени.

```
def calc_sum(a, b):  
    print(f'a is {a}')  
    print(f'b is {b}')  
    result = a + b  
    return result
```

```
calc_sum(3, 5)
```

```
a is 3  
b is 5
```

```
8
```

```
calc_sum(a=3, b=5)
```

```
a is 3  
b is 5
```

```
8
```



## Аргументы функций

**\*args** — это сокращение от «arguments» (аргументы).

```
def calc_sum(*args):  
    result = sum(args)  
    return result
```

```
calc_sum(1)
```

1

```
calc_sum(10, 0, 5)
```

15



## Аргументы функций

**\*\*kwargs** — сокращение от «keyword arguments» (именованные аргументы).

```
def printer(**kwargs):  
    print(type(kwargs))  
    print()  
    for key, value in kwargs.items():  
        print(f'{key} - {value}')
```

```
printer(a=10, b='20')
```

```
<class 'dict'>
```

```
a - 10
```

```
b - 20
```



# Генераторы



# Генератор

**Генератор** - это объект, который сразу при создании не вычисляет значения всех своих элементов.

Генератор хранит в памяти:

1. последний вычисленный элемент
2. правило перехода к следующему
3. условие, при котором выполнение прерывается

Чтобы создать генератор, необходимо определить функцию, как обычно, но использовать **yield** вместо **return**

```
def my_range(start, end):  
    current = start  
    while current < end:  
        yield current  
        current += 1
```



## Итоги урока

- 📌 Узнали, как работать с jupyter notebook эффективней
- 📌 Научились красиво и доступно оформлять jupyter notebook
- 📌 Посмотрели на функционал модуля Random
- 📌 Более детально изучили словари, функции в Python
- 📌 Разобрались с генераторами
- 📌 Познакомились с list, set, dict comprehensions





## Что будет на следующем уроке

- 📌 Анализ данных
- 📌 Подсчет статистик данных
- 📌 Фильтрация данных
- 📌 Сортировки данных







## Полезные ссылки

Модуль Random <https://pythonworld.ru/moduli/modul-random.html>

Jupyter Notebook для начинающих  
<https://webdevblog.ru/jupyter-notebook-dlya-nachinajushhih-uchebnik/>

Добавляем в Jupyter Notebook красоту и интерактивность  
<https://habr.com/ru/post/485318/>

Введение в генераторы Python  
<https://webdevblog.ru/vvedenie-v-generatory-python/>

Что такое \*args и \*\*kwargs в Python?  
<https://habr.com/ru/company/ruvds/blog/482464/>

Что такое list comprehension? Зачем оно? Какие ещё бывают? <https://dvmn.org/encyclopedia/qna/5/chto-takoe-list-comprehension-zachem-ono-kakie-esche-byvajut/>





Спасибо за урок!

