



5-8 классы

Программирование на Python

Презентация занятия

Базы данных. Часть 3.

18 занятие



инжинириум®

МГТУ им. Н.Э. Баумана

2019

СОДЕРЖАНИЕ

1. ВВЕДЕНИЕ. ОРГАНИЗАЦИОННАЯ ИНФОРМАЦИЯ

- Тема занятия
- Цели и задачи занятия
- Результаты занятия
- Материалы для преподавателя
- Материалы для ученика
- Тайминг проведения занятия

2. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

- Запросы SELECT
- Визуализация данных в Python

3. ПРАКТИЧЕСКАЯ ЧАСТЬ

- Визуализация данных из БД
- Библиотека matplotlib
- Библиотека seaborn



ВВЕДЕНИЕ. ОРГАНИЗАЦИОННАЯ ИНФОРМАЦИЯ

Тема: Базы данных. Часть 3.

Цели и задачи:

- Рассказать о существующих форматах представления данных
- Показать, в чем преимущества визуализации данных
- Рассказать про графические библиотеки в Python
- Поработать с библиотекой matplotlib и seaborn

По результатам занятия слушатель будет знать:

- Как писать сложные запросы SELECT
- Каким способом эффективнее всего представлять данные пользователям
- Как визуализировать табличные данные в Python
- Возможности библиотеки matplotlib



Тема: Базы данных. Часть 3.

По результатам занятия слушатель будет уметь:

- Работать с графикой в Python
- Писать сложные запросы SELECT
- Использовать базовые методы в matplotlib
- Использовать базовые методы в seaborn

Тайминг занятия

Таб.1

№	Этапы	время	Сумма
1	Выборка данных из таблиц	25 мин.	25 мин.
2	Представление данных	20 мин.	20 мин.
3	<i>Перерыв</i>	5 мин.	5 мин.
4	Визуализация данных в Python	40 мин.	40 мин.

Тема: Базы данных. Часть 3.

1. ПРИВЕТСТВЕННОЕ СЛОВО ПРЕПОДАВАТЕЛЯ



Тема: Базы данных. Часть 3.

1. ВЫБОРКА ДАННЫХ

1.1 Структура оператора

```
SELECT
  [DISTINCT | DISTINCTROW | ALL]
  select_expression,...
FROM table_references
[WHERE where_definition]
[GROUP BY {unsigned_integer | col_name | formula}]
[HAVING where_definition]
[ORDER BY {unsigned_integer | col_name | formula} [ASC | DESC], ...]
```

- WHERE задает ограничение на строки (!)
- GROUP BY позволяет группировать результаты при выборке из базы данных
- HAVING позволяет фильтровать результат группировки, сделанной с помощью команды GROUP BY
- ORDER BY задает критерии сортировки строк (!)



Тема: Базы данных. Часть 3.

1.2 WHERE

Предикаты условия поиска:

AND (и), **OR** (или) и **NOT**(нет) и скобки, указывающие требуемый порядок вычислений

Предикаты сравнения:

=, <>(не равно), >, >=, < и <=

Примеры:

SELECT * FROM T WHERE C1 = 1;

SELECT * FROM workers WHERE otдел = 40;

SELECT * FROM workers WHERE otдел <> 50;

SELECT name, dolzhnost, otдел FROM workers WHERE otдел = 30 AND dolzhnost="manager";



Тема: Базы данных. Часть 3.

1.2 WHERE

LIMIT

Используется чтобы ограничить количество выбранных записей

Предикат BETWEEN

<Проверяемое выражение> [NOT] BETWEEN <Начальное выражение>
AND <Конечное выражение>

Примеры:

SELECT fam **FROM** student **WHERE** ball **BETWEEN** 81 **AND** 91

SELECT * **FROM** news **WHERE** status='Важные новости' **LIMIT** 1;

SELECT * **FROM** replies **WHERE** author!='Вася' **ORDER BY** date **DESC**
LIMIT 5, 30;



Тема: Базы данных. Часть 3.

1.3 ORDER BY

- является единственным способом отсортировать результирующий набор строк

Упорядочить по возрастанию:

```
SELECT Name, Job, Years FROM Staff WHERE Dept=84 ORDER BY Years  
ASC
```

Упорядочить по убыванию:

```
SELECT Name, Job, Years FROM Staff WHERE Dept=84 ORDER BY Years  
DESC
```



Тема: Базы данных. Часть 3.

Задания

1. Вывести все поля из таблицы Студенты
2. Вывести поля Имя, Фамилия и Отчество из таблицы Студенты
3. Вывести первые 30 полей из таблицы Студенты, возраст которых составляет 15 лет
4. Вывести все поля из таблицы Классы, кроме 5А и которые имеют больше 25 студентов в группе
5. Вывести все поля из таблицы Классы, в которых количество студентов от 20 до 27
6. Вывести первые 5 полей, начиная с 7, из таблицы Классы, кроме тех, где учитель Шатилова Татьяна Николаевна
7. Вывести фамилию, имя и отчество из таблицы Студенты тех учеников, кому еще не исполнилось 18 лет

Тема: Базы данных. Часть 3.

2. ВИЗУАЛИЗАЦИЯ ДАННЫХ

2.1 Лучше один раз увидеть

Визуальная информация запоминается и воспроизводится быстрее. Этот феномен называется **эффект превосходства образа**.

https://www.youtube.com/watch?time_continue=32&v=cLLDUyy8utY&feature=emb_logo

«Если данные подаются в устной форме, человек запоминает из них около 10 процентов, как показывают тесты, проводимые 72 часа спустя. В случае с изображениями данный показатель повышается до 65 процентов.»

2.2 Для чего мы визуализируем данные в БД?

- анализировать данные и их состояние
- накладывать разумные ограничения

Тема: Базы данных. Часть 3.

3. ГРАФИКА В PYTHON

3.1 Matplotlib

Библиотека matplotlib - это библиотека двумерной графики для языка программирования python с помощью которой можно создавать высококачественные рисунки различных форматов.

3.2 Установка

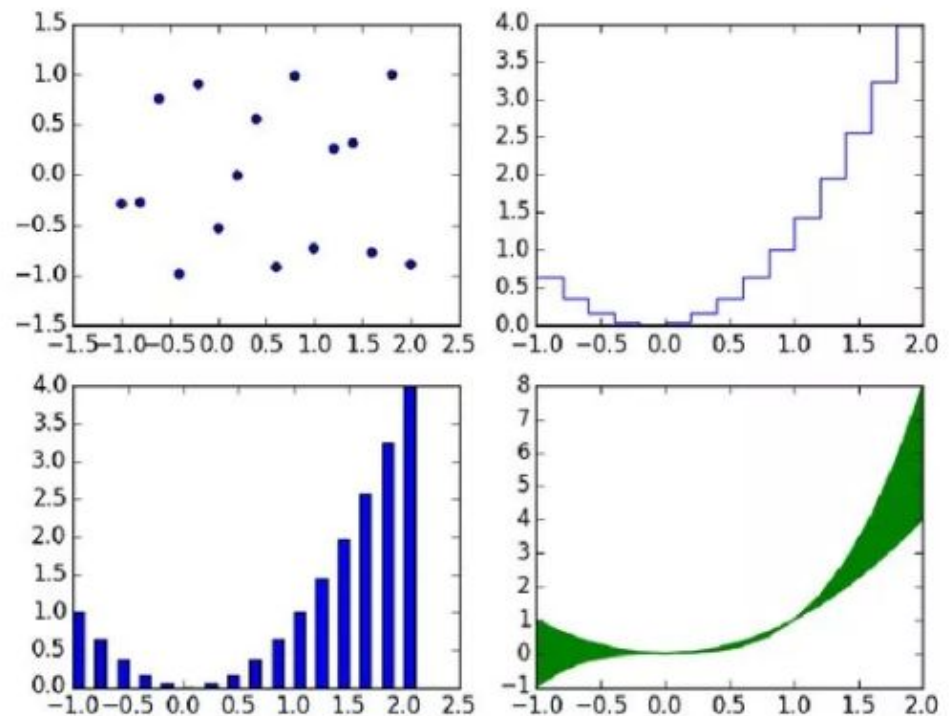
```
> virtualenv PRG1  
> PRG1\Scripts\activate.bat  
  
> pip install matplotlib
```

Тема: Базы данных. Часть 3.

3.3 Модули в Matplotlib

Pyplot - это модуль в Matplotlib (интерфейс).
Функции Pyplot вносят некоторые изменения в график.

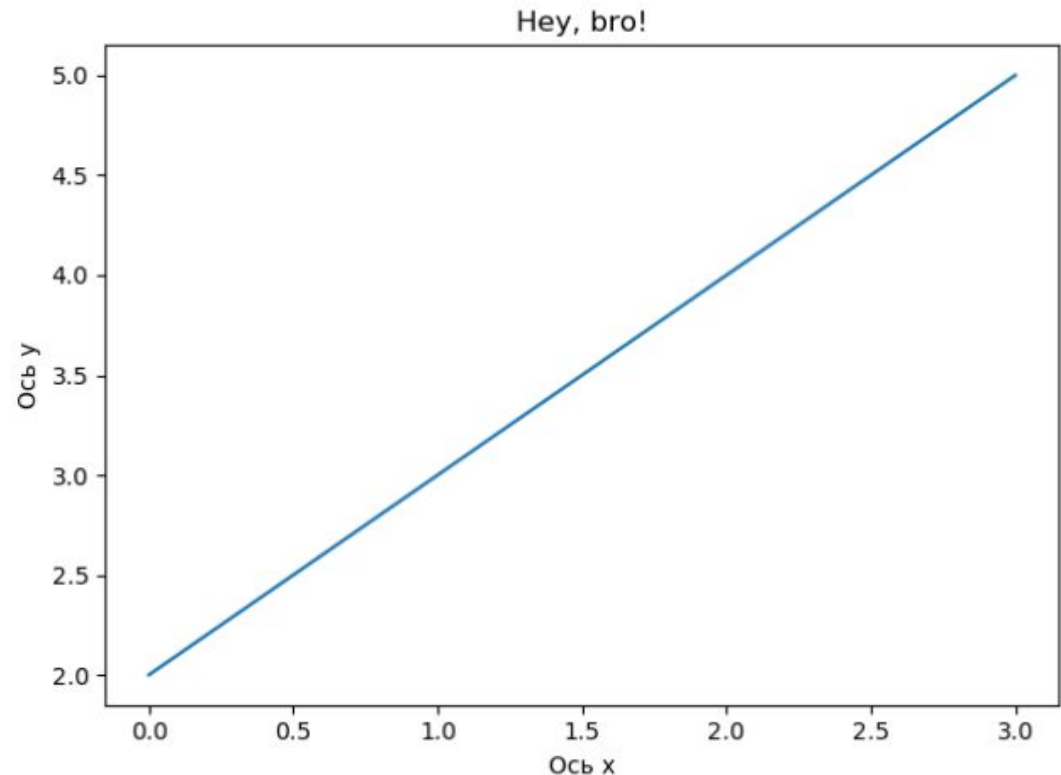
Например, Pyplot может создать график и выделить область построения для этого графика, а затем построить ряд линий в области построения графика.



Тема: Базы данных. Часть 3.

3.4 Построение графиков

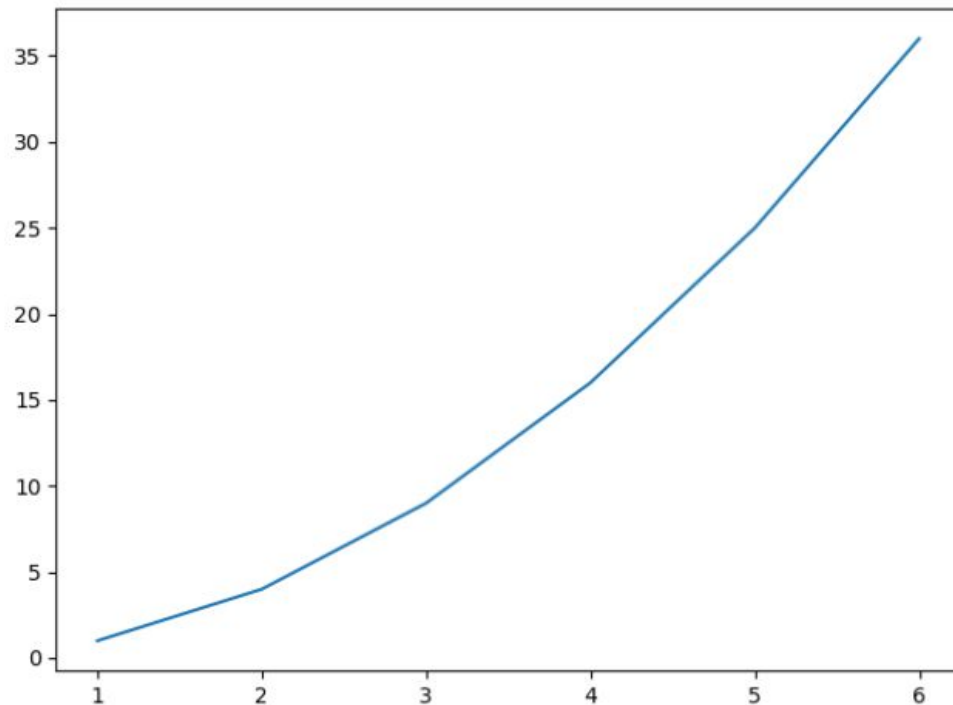
```
import matplotlib.pyplot as plt  
plt.plot([2, 3, 4, 5])  
plt.title('Hey, bro!')  
plt.ylabel('Ось y')  
plt.xlabel('Ось x')  
plt.show()
```



Тема: Базы данных. Часть 3.

3.4 Построение графиков

```
x = [1,2,3,4,5,6]
y = [i**2 for i in x]
plt.plot(x,y)
plt.show()
```

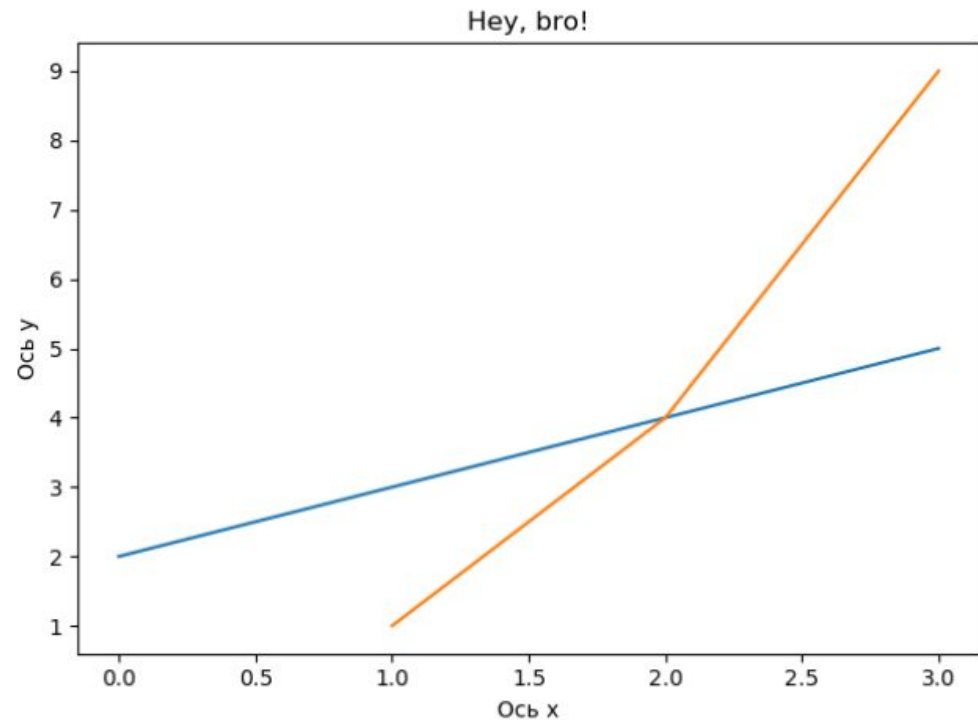


Тема: Базы данных. Часть 3.

3.4 Построение графиков

Графики могут накладываться друг на друга:

```
plt.plot([2, 3, 4, 5])  
plt.title('Hey, bro!')  
plt.ylabel('Ось y')  
plt.xlabel('Ось x')  
# plt.show()  
# graphics  
x = [1,2,3]  
y = [i**2 for i in x]  
plt.plot(x,y)  
plt.show()
```

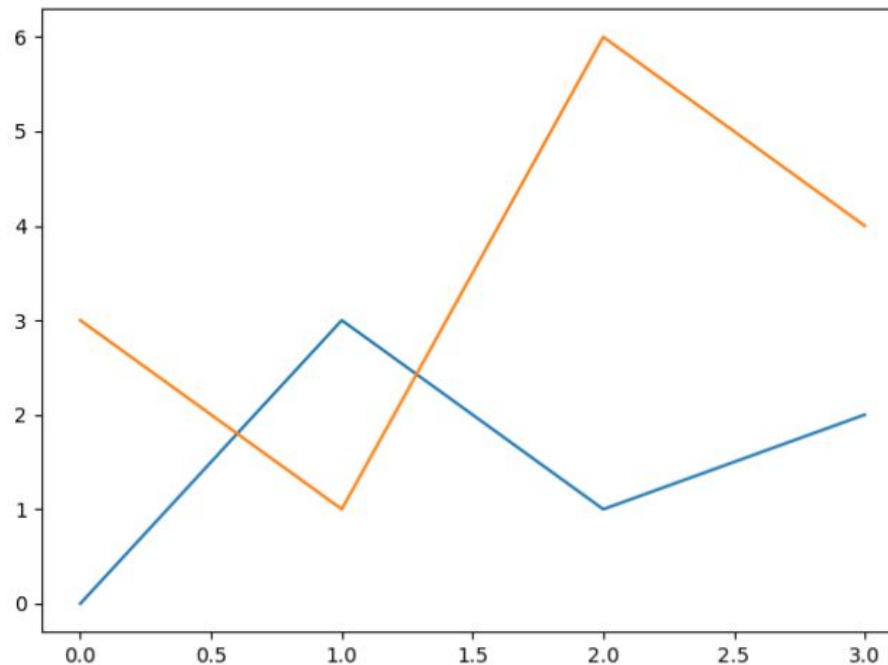


Тема: Базы данных. Часть 3.

3.4 Построение графиков

Plot() является универсальной командой и принимает произвольное количество аргументов:

```
plt.plot([0, 1, 2, 3], [0, 3, 1, 2], [3, 1, 6, 4])  
plt.show()
```



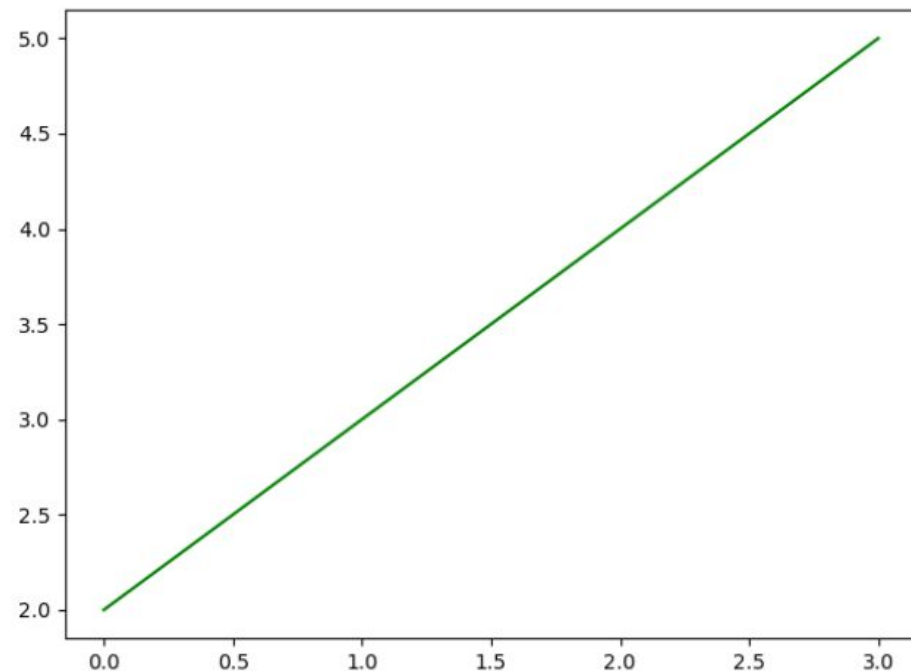
Тема: Базы данных. Часть 3.

3.5 Настройка внешнего вида графика

Тип и цвет линии

```
data = [2, 3, 4, 5]
plt.plot(data, color = 'g')
# plt.plot(data, '--')
# plt.plot(data, '*')
# plt.plot(data, 'g*')
plt.show()
```

Что покажут остальные
строчки кода?

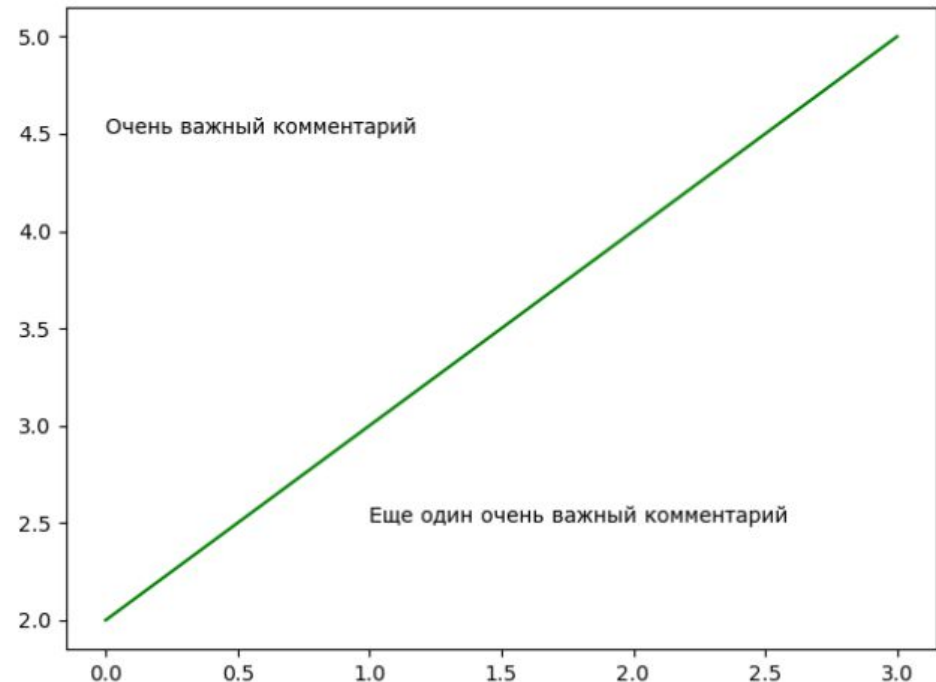


Тема: Базы данных. Часть 3.

3.5 Настройка внешнего вида графика

Надписи на графике

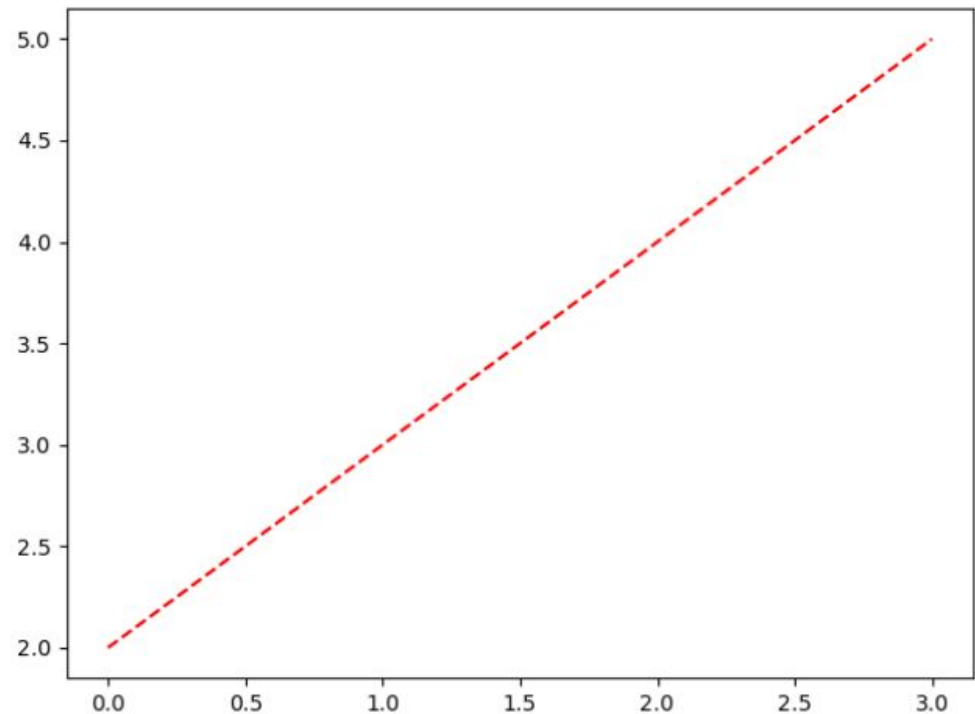
```
data = [2, 3, 4, 5]
plt.plot(data, color='g')
# plt.plot(data, '--')
# plt.plot(data, '*')
# plt.plot(data, 'g*')
plt.text(0, 4.5, 'Очень важный комментарий', family="verdana")
plt.text(1, 2.5, 'Еще один очень важный комментарий', family="verdana")
plt.show()
```



Тема: Базы данных. Часть 3.

3.6 Как сохранить график

```
data = [2, 3, 4, 5]  
plt.plot(data, 'r|--')  
plt.savefig('saved_plot.png')  
plt.show()
```



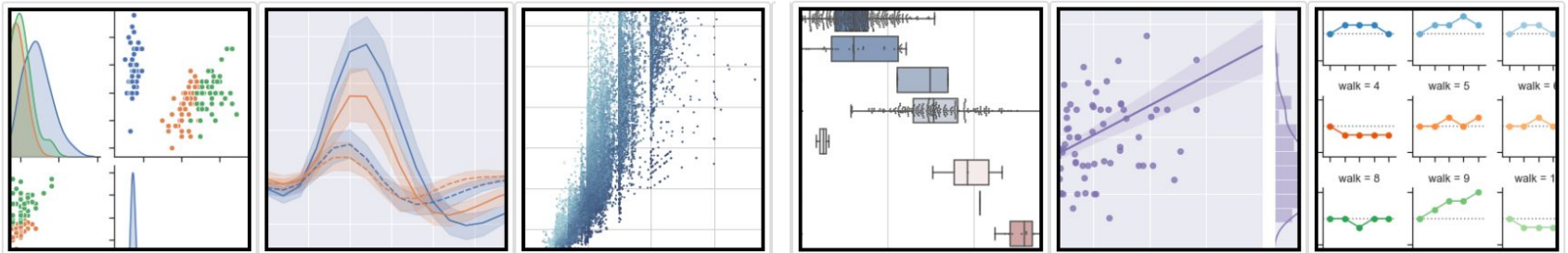
Тема: Базы данных. Часть 3.

4. Seaborn

Seaborn – это надстройка над matplotlib

Новое (на самом деле уже не новое) средство визуализации, которое гибко подстраивается под табличные данные с типами от pandas и numpy.

Обладает несколькими приятными дополнениями к графикам и встроенными эстетичными темами.



Тема: Базы данных. Часть 3.

4.1 Загрузка данных

```
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
import sqlite3

conn = sqlite3.connect('school.db')

table = pd.read_sql("SELECT * FROM students", conn)
table.info()
print(table)
# print(table.head())
```

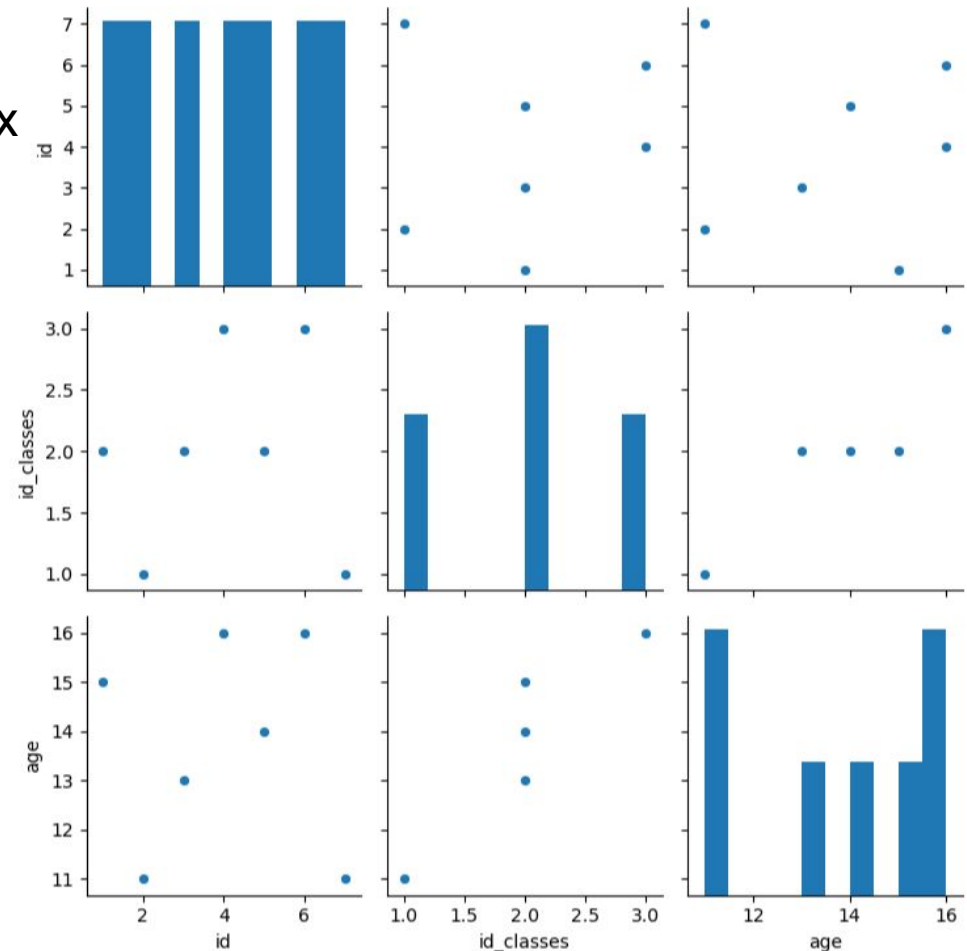


Тема: Базы данных. Часть 3.

4.2 pairplot

pairplot показывает отношения между всеми парами переменных

```
sns.pairplot(table)  
plt.show()
```



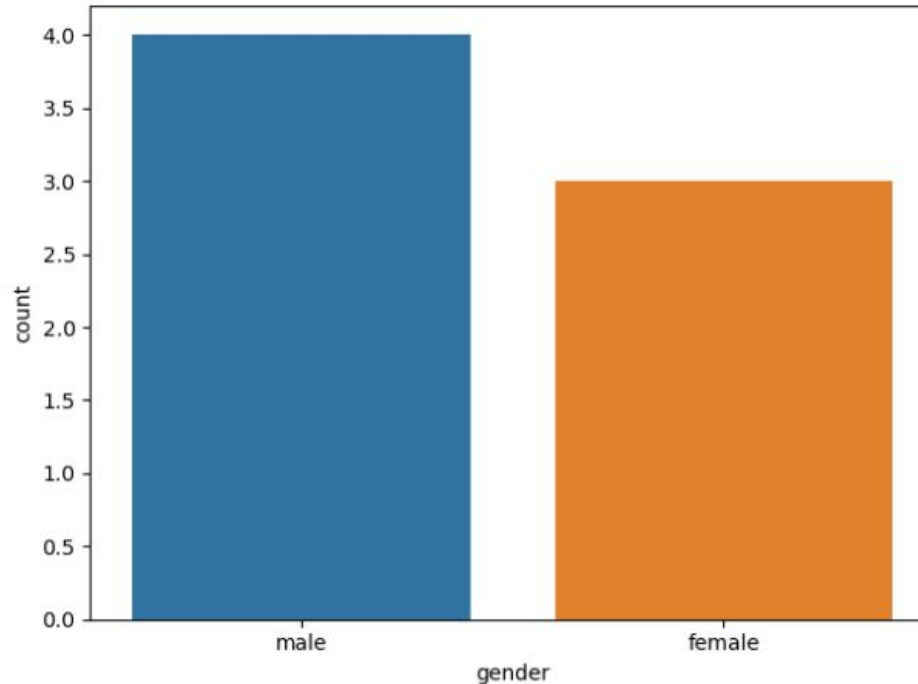
Тема: Базы данных. Часть 3.

4.3 countplot

С помощью функции **countplot** удобно визуализировать количество по категориям

(в таблицу students необходимо добавить поле gender)

```
print(table['gender'].value_counts())  
sns.countplot(x='gender', data=table)  
plt.show()
```

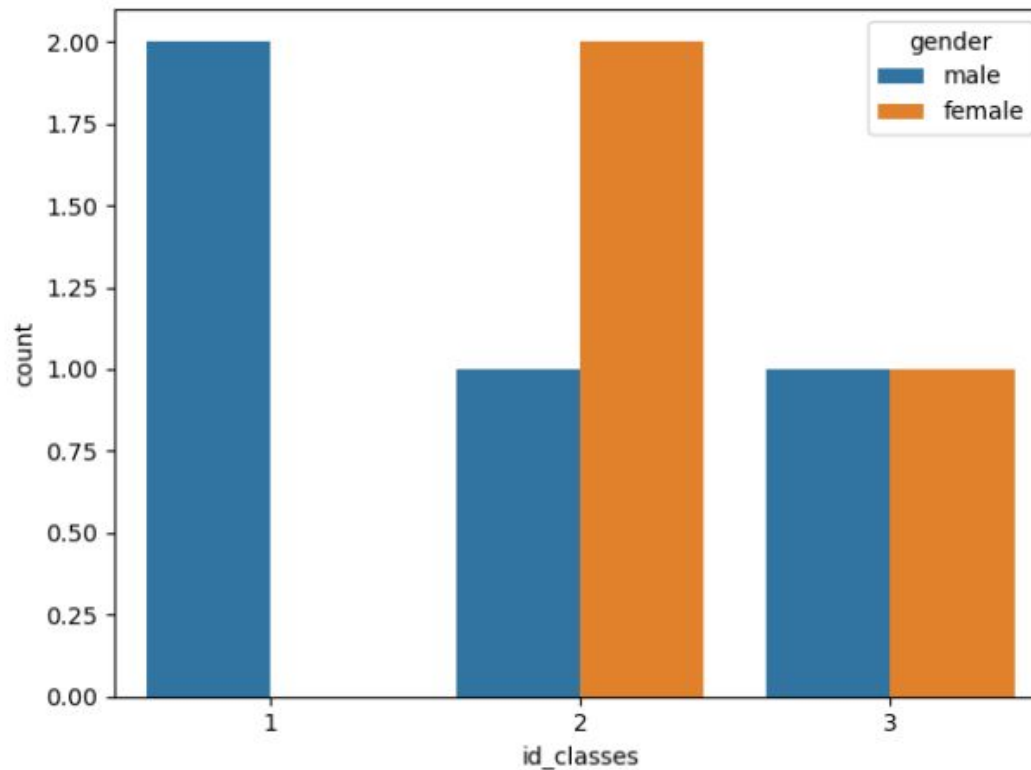


Тема: Базы данных. Часть 3.

4.3 countplot

Посмотрим, как распределен пол по классам

```
sns.countplot(x='id_classes', data=table, hue='gender')  
plt.show()
```



Тема: Базы данных. Часть 3.

Задания

1. Посчитайте количество человек по возрастной категории в таблице Студенты
2. Покажите отношения между всеми парами переменных таблицы Классы
3. Распечатайте первые 5 записей таблицы Оценки
4. Посмотрите, как распределены оценки у студентов по их айди



Тема: Базы данных. Часть 3.

Рефлексия

1. Что мы сегодня узнали?
2. Чему научились?
3. Зачем нужно визуализировать данные?
4. С какими библиотеками познакомились?