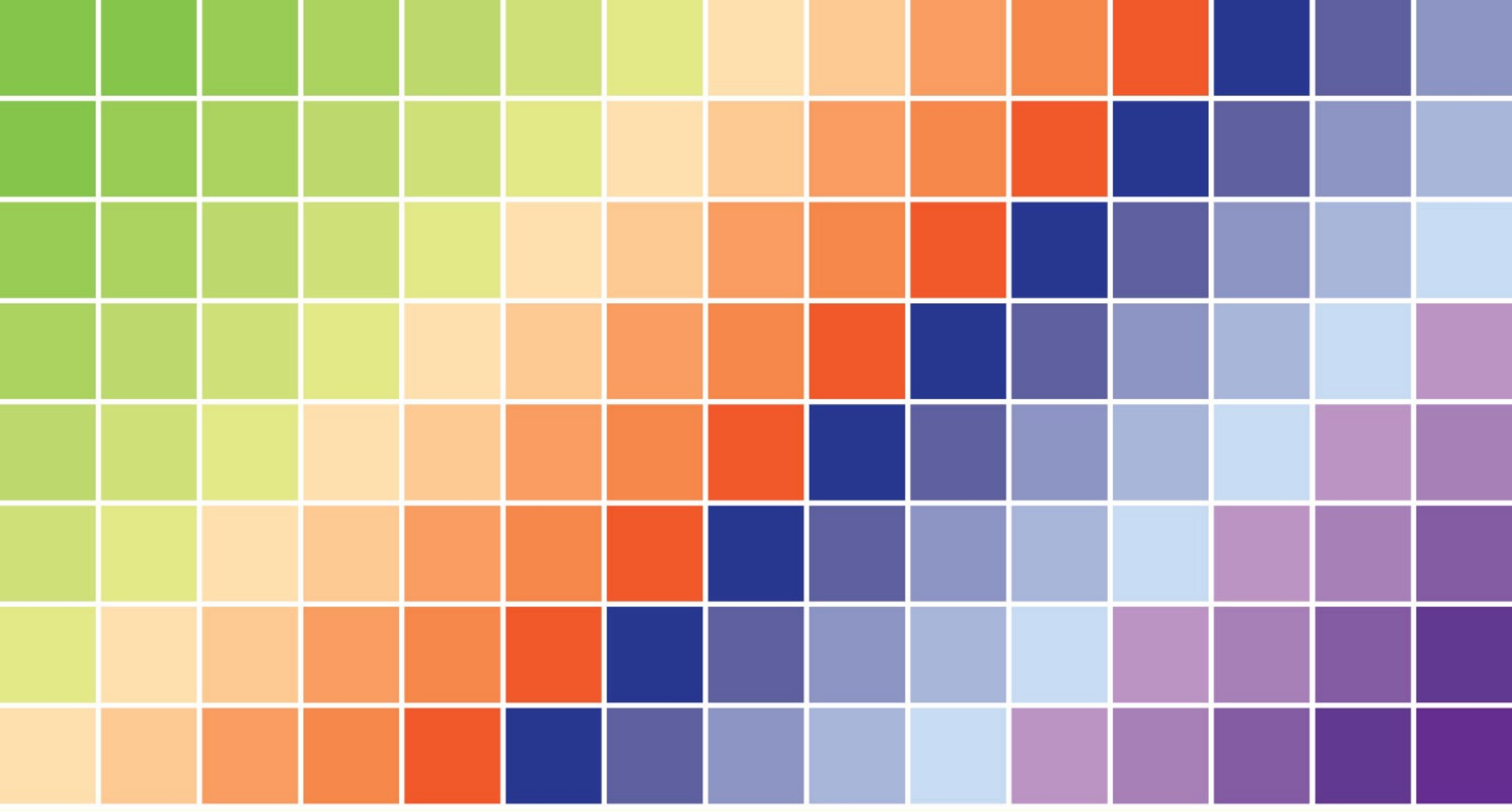


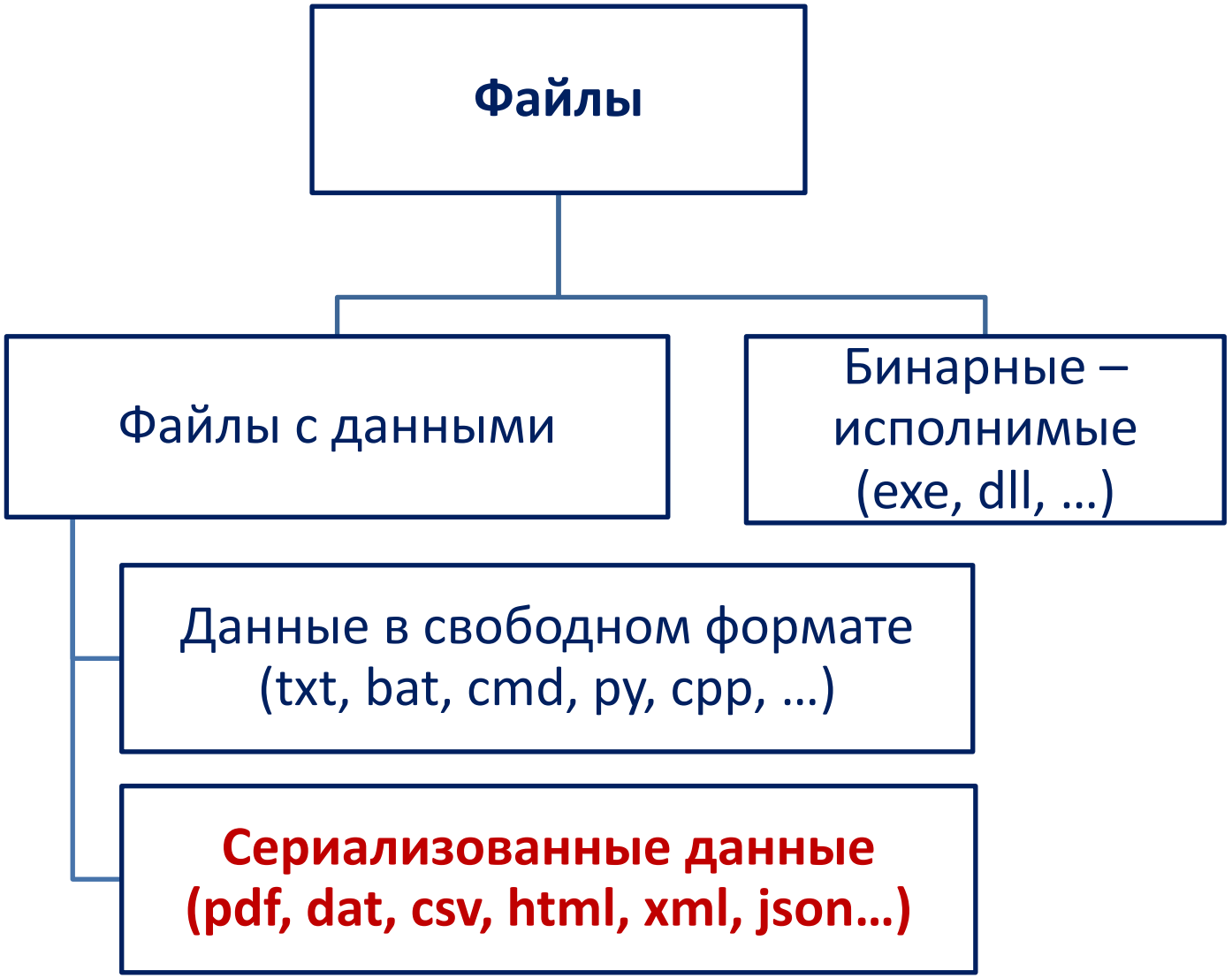
School of Computer Science

ИСПОЛЬЗОВАНИЕ МОДУЛЕЙ. ПРАКТИЧЕСКИЕ ПРИМЕРЫ

ПРОГРАММИРОВАНИЕ НА PYTHON

Лекции для IT-школы

ФАЙЛОВЫЕ ФОРМАТЫ



СЕРИАЛИЗАЦИЯ. ХРАНЕНИЕ СТРУКТУРИРОВАННЫХ ДАННЫХ

- Сериализация (консервация) нужна для передачи и хранения данных во внешней среде (вне программы)
- Модуль pickle – сериализация с последовательным доступом
- Модуль shelve – сериализация с произвольным доступом (как в словаре)
- См. примеры в pickle_it.py и shelve_it.py

РЕЖИМЫ БИНАРНОГО ФАЙЛА ПРИМЕНИМЫ ДЛЯ СЕРИАЛИЗАЦИИ

Режим	Значение
rb	Чтение. Выдается исключение при отсутствии файла
wb	Запись. Если файл существует, то он будет переписан. Если файла нет, то он будет создан
ab	До-запись. Если файл существует, то новые данные будут дописаны в конец. Если файл не существует, то он будет создан
rb+	Чтение и запись. Если файл не существует, то выдается исключение
wb+	Запись и чтение. Если файл существует, то он будет переписан. Если файла нет, то он будет создан
ab+	До-запись и чтение. Если файл существует, то новые данные будут дописаны в конец. Если файла нет, то он будет создан

ИЗБРАННЫЕ ФУНКЦИИ МОДУЛЯ PICKLE

Вызов	Описание
dump(object, file)	Пишет законсервированную версию объекта в файл
load(file)	Восстанавливает очередной консервированный объект из файла и возвращает его
dumps(object)	Возвращает сериализованное представление object в виде набора байт для последующего сохранения или передачи

РЕЖИМЫ ДОСТУПА К ФАЙЛУ-ПОЛКЕ ДЛЯ КОНСЕРВАЦИИ С ПОМОЩЬЮ SHELVE

Режим	Значение
c	Открытие полки на чтение или запись. Если файл полки не существует, он будет создан. Используется как значение по умолчанию
n	Создание нового файла-полки для чтения или записи. Если файл существует, то его содержимое будет заменено
r	Чтение из файла-полки. Если файл не существует, сгенерируется исключение
w	Запись в файл-полку. Если файл не существует, сгенерируется исключение

ИЗБРАННЫЕ МЕТОДЫ ОБЪЕКТА SHELVE
SHELVE – СОХРАНЯЕМЫЙ СЛОВАРЬ

Вызов	Описание
sh.clear()	Удаляет все элементы из полки sh
sh.keys()	Возвращает набор всех ключей полки
sh.pop(k), подобна del sh[k]	pop() возвращает значение ключа k и удаляет его из полки или возбуждает исключение KeyError, если ключ k не найден. del sh[k] ничего не возвращает
sh.values()	Возвращает набор всех значений полки sh
sh.sync()	Аналог file.flush(), сохраняет изменения в файл полки, не закрывая файл
sh.close()	Закрывает полку с сохранением всех изменений в файле

МОДУЛИ

Стандартные – Python 3 Global Module Index – доступны по умолчанию:

- https://docs.python.org/3/py-modindex.html

Дополнительные – Python Package Index:

- https://pypi.org

Установка с помощью команды:

- pip install package_name

ИМПОРТ МОДУЛЕЙ

Пусть **X** – имя того, что идёт после **import.**

1. import <модуль (без .py)>

Если **X** – имя модуля, то для того, чтобы использовать объекты, определённые в **X**, нужно писать **X.объект**.

2. from <модуль (без .py)> import <объект>

Если **Z –** имя объекта, то к ему можно обращаться напрямую без точечной нотации с именем модуля

Опционально после любого выражения ***import X*** можно добавить ***as Y***. Это переименует **X** в **Y** в пределах скрипта. Важно, что имя **X** с этого момента становится недействительным:

```
import numpy as np
np.array([[1, 2, 3], [4, 5, 6]])
```

АВТОМАТИЗАЦИЯ РУТИННЫХ ЗАДАЧ С ПОМОЩЬЮ PYTHON

- Книга: https://automatetheboringstuff.com/
- Видео пробных уроков: здесь
- Инструкция по установке сторонних модулей: здесь

- Автор: Эл Свейгарт

https://www.udemy.com/user/al-sweigart/

Al Sweigart is a software developer in San Francisco. He has written four Python programming books, spoken at Python conferences, and has taught both kids and adults how to program. Python is his favorite programming language, and he is the developer of several open source modules for it. He is driven to make programming knowledge available to all, and his books freely available under a Creative Commons license.

УСТАНОВКА СТОРОННЕГО МОДУЛЯ PYPERCLIP ДЛЯ РАБОТЫ С БУФЕРОМ

- Запустите командный интерпретатор cmd.exe или Power Shell
- Выдайте команду: pip install pyperclip
- После окончания установки модуля в Python IDLE Shell введите код:

```
>>> import pyperclip
>>> pyperclip.copy('Hello world!')
>>> pyperclip.paste()
'Hello world!'
```

БУФЕР ОБМЕНА С КЛЮЧАМИ

- Вызов программы:
  mcb.py save keyword | list | keyword
- План программы:
  - Проверить аргумент командной строки на ключевое слово
  - Если аргумент равен save, то буфер сохраняется под ключом keyword
  - Если аргумент равен list, то все ключевые слова копируются в буфер обмена
  - В противном случае текст, соответствующий ключевому слову, копируется в буфер
- См. пример в mcb/mcb.py

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ
УСОВЕРШЕНСТВОВАНИЕ MCB.PY

- Для удаления одного из сохраненных ключевых слов, поддержите следующий формат : mcb.py delete keyword

- Реализуйте удаление всех ключевых слов командой: mcb.py purge

ВЗАИМОДЕЙСТВИЕ С БРАУЗЕРОМ
МОДУЛЬ WEBBROWSER

- webbrowser:
  - стандартный модуль Python
  - предназначен для открытия браузера на определенной web-странице

- В Python IDLE Shell введите код:

```
>>> import webbrowser
>>> webbrowser.open("www.compassplus.com")
True
```

ПРАКТИЧЕСКИЙ ПРИМЕР «АДРЕС НА КАРТЕ»

- Программа должна:
  - Получить почтовый адрес из командной строки или буфера обмена
  - Открыть Google Maps на странице, соответствующей указанному адресу

- Код программы map/map_it.py:
  - Читает аргументы командной строки
  - Читает содержимое буфера обмена
  - Вызывает функцию webbrowser.open() для нужной страницы

ПОЛУЧЕНИЕ КАРТЫ МЕСТНОСТИ
ПРЕИМУЩЕСТВА СКРИПТА MAP_IT.PY

Ручной вариант	Использование map_it.py
1) Выделение адреса	1) Выделение адреса
2) Копирование адреса	2) Копирование адреса
3) Открытие браузера	3) Запуск скрипта map_it.py
4) Переход по адресу https://www.google.ru/maps	–
5) Щелчок в поле ввода адреса	–
6) Вставка адреса	–
7) Нажатие Enter	–

МАТЕМАТИЧЕСКИЕ ОПЕРАЦИИ
NUMPY

NumPy это open-source модуль, который предоставляет общие математические и числовые операции (https://docs.scipy.org/doc/)

NumPy (Numeric Python) предоставляет базовые методы для работы массивами и матрицами большой размерности.
SciPy (Scientific Python) это расширение numpy, содержащее большое количество прикладных процедур, таких как минимизация, преобразование Фурье, регрессия и т.д.

В совокупности (и вместе с matplotlib) модули способны заменить математические пакеты (MatLab)

МАТЕМАТИЧЕСКИЕ ОПЕРАЦИИ
NUMPY

Основные объекты – векторы и матрицы, представленные в виде массива *array*.

- Похожи на списки
- **НО:**
- Это «плоские» массивы, а не контейнерные последовательности, то есть:
  - Элементы массива должны быть однотипными (float или int)
- Числовые операции выполняются быстрее и эффективнее

ОСНОВНЫЕ ФУНКЦИИ NUMPY
СОЗДАНИЕ ОБЪЕКТОВ

Функция	Описание
array() array([[1, 2, 3], [4, 5, 6]], complex)	Преобразовать объект (обычно список) в массив. Можно указать тип объектов (int, float, complex)
arange() arange(1, 5, 0.5)	Аналогична функции range() из стандартной библиотеки Python. Позволяет строить вектор с указанием шага в виде десятичной дроби
matrix() matrix([[1, 2], [3, 4]]) matrix('[1, 2; 3, 4]')	Создание матрицы. В отличие от объекта array могут быть только двумерными. Аргументом может быть список или строка, содержащая элементы матрицы через ;
zeros() zeros((3, 4))	Создание нулевой матрицы
eye() eye(3)	Создание единичной матрицы – «кругом нули, а по диагонали единицы»

ДОСТУП К ЭЛЕМЕНТАМ

Доступ к элементам массива осуществляется по целочисленным индексами, начинается отсчет с 0:

```
>>> table = np.array([[1, 2, 3], [4, 5, 6]])
>>> table[1][1]
>>> 5
>>> table[1,1]
>>> 5
```

Доступ к подмассиву:

```
>>> table[1]
>>> array([4, 5, 6])
```

Недостающие индексы можно явно заменить списком всех возможных индексов вдоль соответствующей оси с помощью знака «:» – символ среза:

```
>>> table[1, :]
>>> array([4, 5, 6])

>>> table[:, 1]
>>> array([2, 5])
```

ПРЕОБРАЗОВАНИЕ ОБЪЕКТОВ

Функция	Описание
array.ravel()	Преобразование матрицы в одномерный вектор
np.where()	Возвращает один из двух заданных элементов в зависимости от условия
array.transpose()	Транспонирование матрицы
array.reshape()	Поменять размеры вдоль осей или размерность. Важно, чтобы количество элементов сохранилось

МАТЕМАТИЧЕСКИЕ ОПЕРАЦИИ

Если A и B массивы **одинакового размера**, то их можно:

- складывать,
- умножать,
- вычитать,
- делить,
- возводить в степень.

Эти операции выполняются **поэлементно**.

Матричное умножение по правилам линейной алгебры выполняется с помощью функции *dot()* или операции @

ПОСТРОЕНИЕ ГРАФИКОВ
MATPLOTLIB

matplotlib – это библиотека двумерной графики, с помощью которой можно создавать высококачественные рисунки различных форматов.

Главной единицей (объектом самого высокого уровня) при работе с matplotlib является рисунок (Figure). Любой рисунок в matplotlib имеет вложенную структуру:

Figure (Рисунок) -> Axes (Область рисования) -> Axis (Координатная ось)

ИЕРАРХИЧЕСКАЯ СТРУКТУРА РИСУНКА

- **Рисунок (Figure)**

Является объектом самого верхнего уровня, на котором располагаются одна или несколько областей рисования (Axes), элементы рисунка Artisits (заголовки, легенда и т.д.) и основа-холст (Canvas).

- **Область рисования (Axes)**

Часть изображения с пространством данных. Каждая область рисования содержит две координатных оси, которые упорядочивают отображение данных.

- **Координатная ось (Axis)**

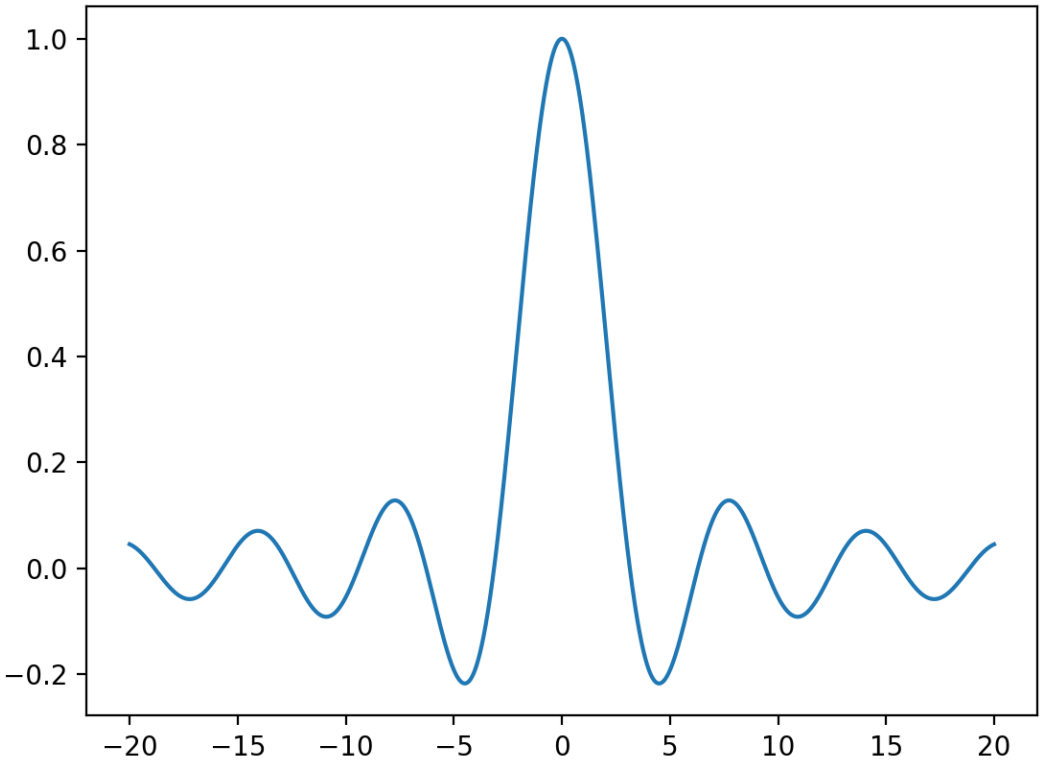
Определяет область изменения данных, на неё наносятся деления ticks и подписи к делениям ticklabels. Расположение делений определяется объектом Locator, а подписи делений обрабатывает объект Formatter. Конфигурация координатных осей заключается в комбинировании различных свойств объектов Locator и Formatter.

- **Элементы рисунка (Artists)**

Элементы рисунка Artists являются как бы красной линией для всех иерархических уровней. Практически всё, что отображается на рисунке является элементом рисунка (Artist), даже объекты Figure, Axes и Axis. Элементы рисунка Artists включают в себя такие простые объекты как текст (Text), плоская линия (Line2D), фигура (Patch) и другие.

ПОСТРОЕНИЕ ГРАФИКОВ

Пример построения в *matplottest.py*



Хорошая статья с примерами построения различных графиков и диаграмм:
https://habr.com/ru/post/468295/

СПАСИБО ЗА ВНИМАНИЕ !

ВОПРОСЫ ?

School of Computer Science