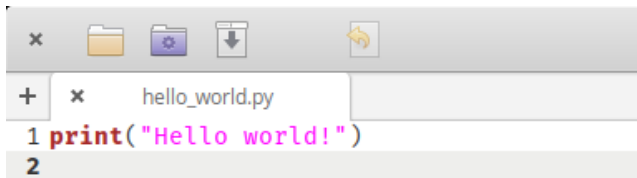


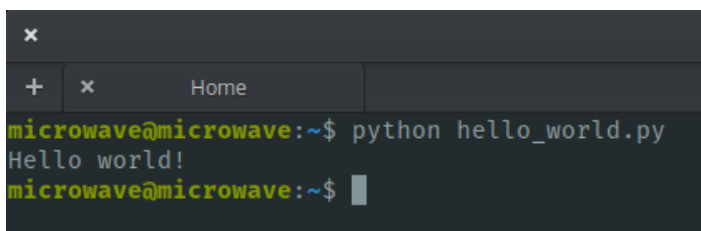
# Начало работы в Jupyter Notebook

Программы на языке Python можно запускать двумя основными способами.

1. Можно написать программу целиком, сохранить её в файле или нескольких файлах, а затем запустить её целиком и получить результат её работы:

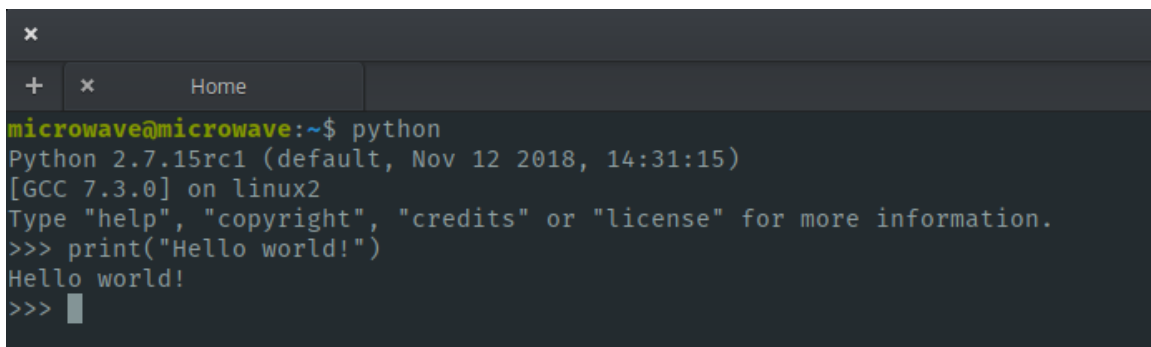


```
1 print("Hello world!")
2
```



```
microwave@microwave:~$ python hello_world.py
Hello world!
microwave@microwave:~$
```

2. Можно запустить интерактивную консоль, и вводить в ней команды по одной, получая результат выполнения каждой команды отдельно:

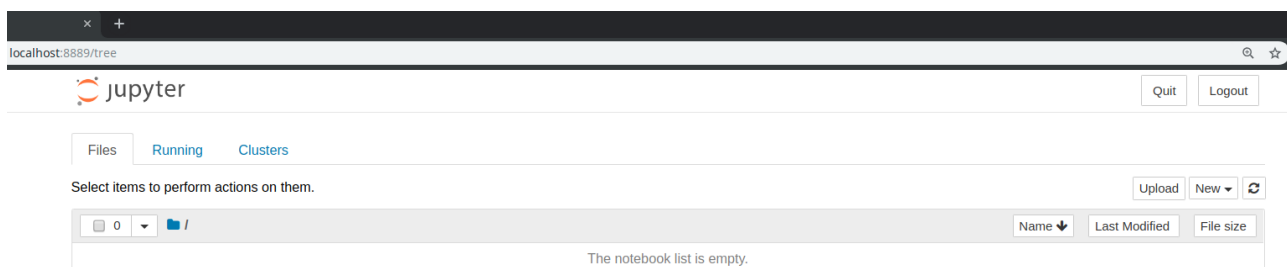


```
microwave@microwave:~$ python
Python 2.7.15rc1 (default, Nov 12 2018, 14:31:15)
[GCC 7.3.0] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello world!")
Hello world!
>>>
```

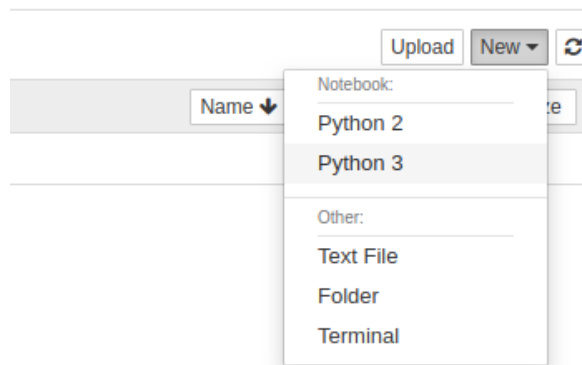
В Data Science, как правило, более удобным оказывается именно второй вариант, поскольку работа с данными - это, как правило, исследовательская работа. Она не всегда нацелена на написание конкретной программы.

У обычной интерактивной консоли есть очень удобный аналог - Jupyter Notebook. Мы будем использовать Jupyter Notebook в составе дистрибутива Anaconda, который можно скачать с [официального сайта](https://www.anaconda.com/) (<https://www.anaconda.com/>).

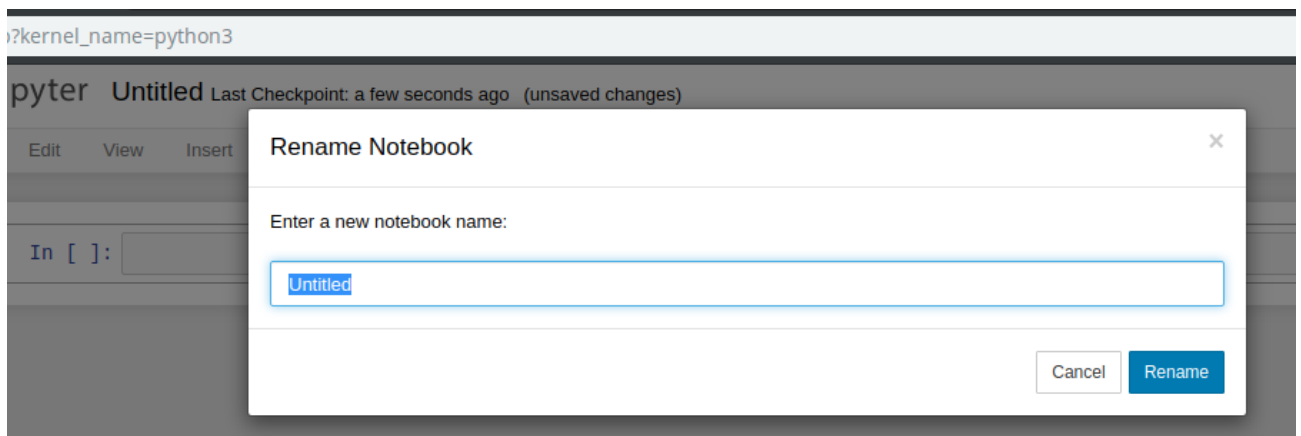
После установки и запуска мы оказываемся в домашней директории - это та папка, из которой был запущен Jupyter Notebook.



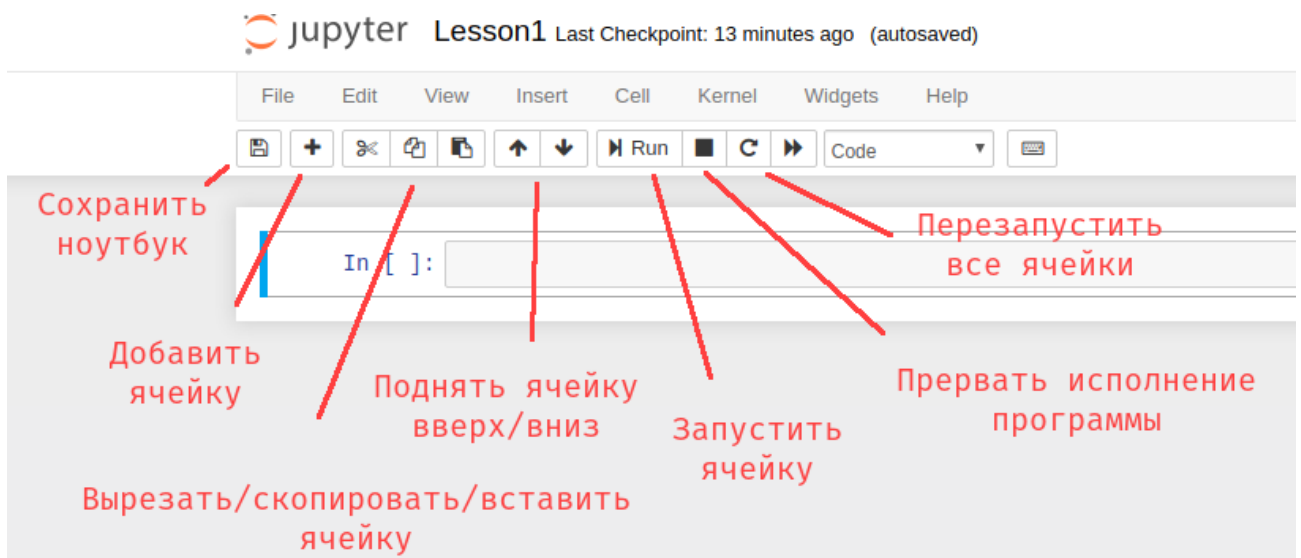
Нажимаем меню **New** и выбираем в нём **Python 3** чтобы открыть создать новый ноутбук:



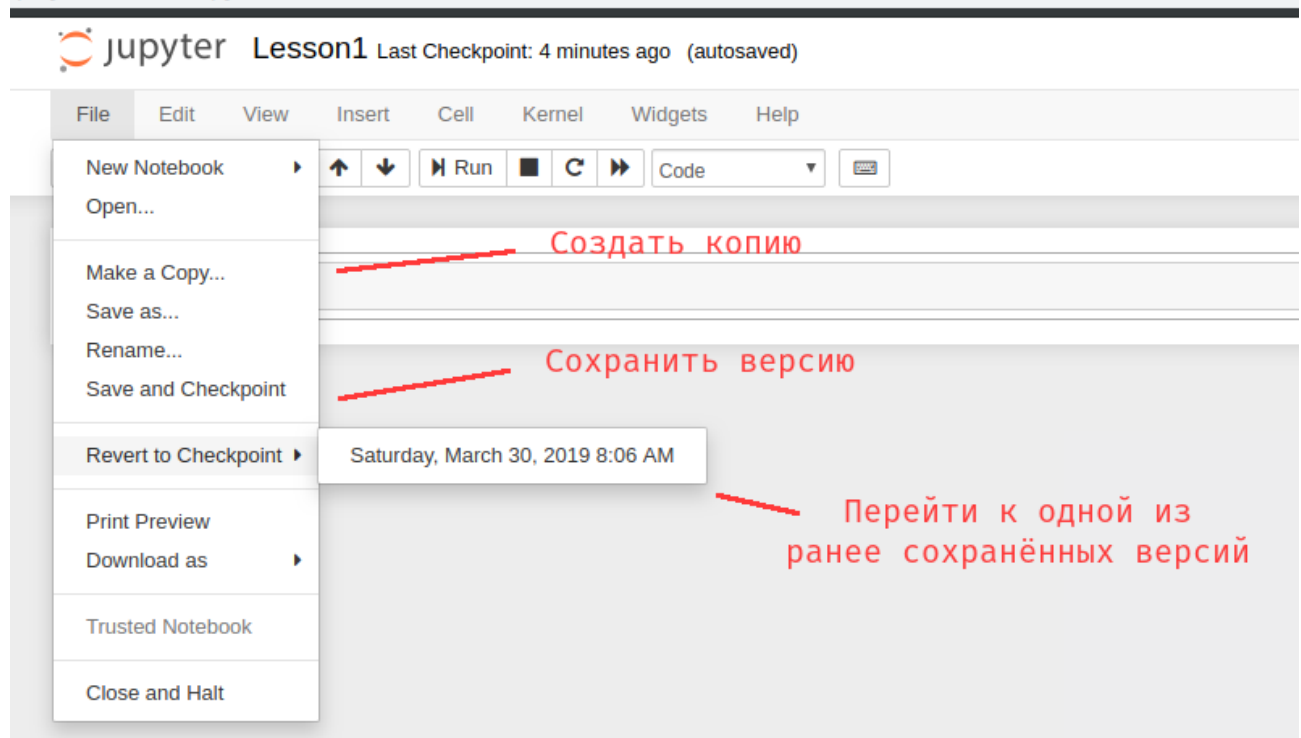
Нажав на название нашего ноутбука в шапке (сейчас он называется `Untitled`), можно его переименовать. Переименуем ноутбук в `Lesson1`.



В верхней части расположен тулбар, позволяющий управлять нашим ноутбуком. Вот некоторые из доступных через него функций:



В меню `File` кроме интуитивно понятных пунктов есть несколько очень важных опций:

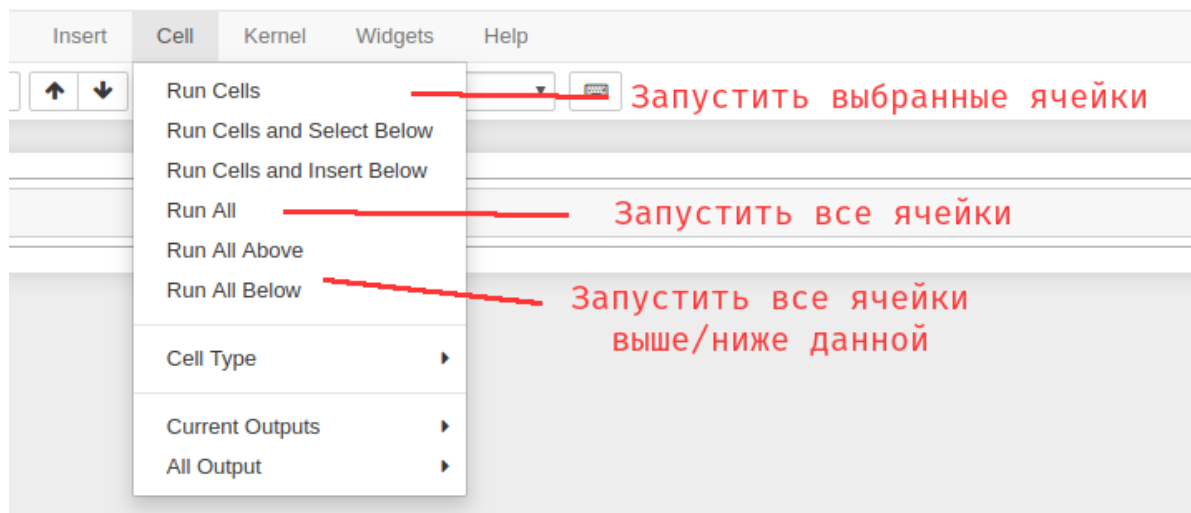


В меню **Edit** содержатся действия над ячейками. Тут можно вырезать, копировать, вставлять ячейки (как сверху, так и снизу от данной), удалять. Также можно разделять ячейки или наоборот сливать в одну, ячейки можно двигать вверх и вниз. В этом же меню находится очень полезная функция **Find and Replace**, которая позволяет выполнять поиск и замену по содержимому ячеек в нутбук.

С помощью меню **Insert** можно создавать новые ячейки в нутбук, причём как выше данной, так и ниже.

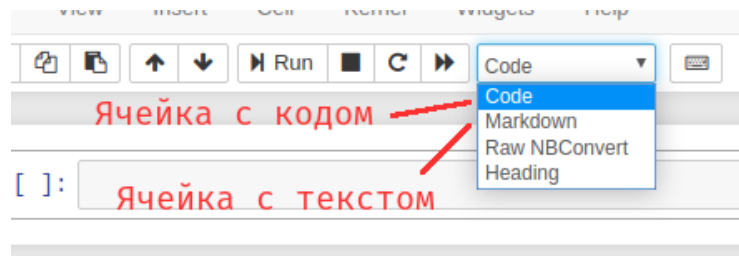
В меню **Cell** располагаются опции, связанные с запуском ячеек:

Lesson1 Last Checkpoint: 26 minutes ago (autosaved)



С помощью меню **Kernel** можно управлять процессами, происходящими в нашем нутбук. С помощью опции **Interrupt** можно прервать текущий процесс (это же можно сделать и с помощью определённой кнопки на тулбаре), с помощью **Restart** можно полностью перезагрузить нутбук, а с помощью **Shutdown** можно просто его полностью остановить.

В тулбаре также есть выпадающее меню, в котором можно выбрать тип ячейки. Здесь мы будем использовать в основном две возможности - **Code** и **Markdown**. **Code** - тип ячейки по умолчанию. В таких ячейках можно набирать код программы и запускать его. **Markdown** - ячейка с текстом. Здесь можно набирать текст, используя синтаксис [Markdown](https://ru.wikipedia.org/wiki/Markdown) (<https://ru.wikipedia.org/wiki/Markdown>), а также [LaTeX](https://ru.wikipedia.org/wiki/LaTeX) (<https://ru.wikipedia.org/wiki/LaTeX>).



## Горячие клавиши

Jupyter Notebook позволяет также использование горячих клавиш, чтобы упростить и ускорить работу в нём. Например, вот несколько способов быстро запустить текущую ячейку:

- Shift + Enter - запустить ячейку и перейти в следующую
- Ctrl + Enter - запустить ячейку и остаться в ней

Список горячих клавиш можно найти в меню Help → Keyboard Shortcuts .

В режиме написания кода в ячейке также полезно использовать клавишу Tab : она выводит подсказку о том, какие команды начинаются с данных букв. Например, можно ввести в ячейке pr , затем нажать Tab , и нам выпадет окно со всеми командами, которые начинаются с pr .

Если вы хотите узнать подробности о том, что делает какая-то функция, какие аргументы она принимает и пр. - вот несколько способов вызвать документацию к функции:

- Набрать имя этой функции (например, print ) и нажать Shift + Tab
- Поставить знак вопроса перед этой функцией (например, ?print ) и запустить данную ячейку

В ячейке можно также запускать команды из командной строки вашей операционной системы, для этого надо поставить символ ! перед запускаемой командой. Например, в операционной системе Windows команда !dir выведет на экран содержимое папки, в которой вы находитесь.

## Магические команды

В Jupyter Notebook можно запускать так называемые *магические команды*. Список таких команд можно посмотреть следующим образом: набрать в ячейке %lsmagic и запустить ячейку. Команды, которые начинаются с одного знака % , действуют только на одну строку кода, а те, которые начинаются с двух знаков, действуют на всю ячейку.

Вот некоторые магические команды:

- %run script.py - запускает код из файлов с расширением .py или из других ноутбуков
- %load script.py - загружает код из файла внутрь данной ячейки, а сама команда %load становится закомментированной
- %runcat script.py - загружает код из файла во всплывающее окно внизу страницы

```
In [ ]: %load hello_world.py
```

```
In [ ]: # %load hello_world.py
print("Hello world!")
```

- %%time - показывает, за какое время исполнились команды из данной ячейки

In [1]: `%%time`

```
for i in range(100000):  
    a = 0
```

CPU times: user 4.24 ms, sys: 8 µs, total: 4.25 ms

Wall time: 4.07 ms

- `%%timeit` - запускает содержимое данной ячейки несколько раз и считает среднее время выполнения

In [2]: `%%timeit`

```
10 ** 4
```

7.65 ns ± 0.0689 ns per loop (mean ± std. dev. of 7 runs, 100000000 loops each)