

Лекція №1

Опис, встановлення та основи роботи з Jupiter Notebook

Jupyter Notebook - неймовірно потужний інструмент для інтерактивної розробки та подання проектів в області наук про дані. У цій лекції ви дізнаєтеся, як налаштувати Jupyter Notebooks на локальному комп'ютері і як почати використовувати його в ваших проектах.

Почнемо з визначення: що таке «notebook» (блокнот)? Блокнот об'єднує код і його висновки в єдиний документ, який об'єднує візуалізацію, розповідний текст, математичні рівняння та інші мультимедійні. Цей інтуїтивно зрозумілий робочий процес сприяє ітеративній і швидкій розробці, що робить ноутбуки все більш популярним вибором для подання в даних і їх аналізу.

Проект Jupyter є наступником більш раннього проекту IPython Notebook, який вперше був опублікований в якості прототипу в 2010 році. Хоча в Jupyter Notebooks можна використовувати з багатьма різними мовами програмування, саме Python, є найбільш поширеним варіантом використання.

Встановлення:

Найпростіше почати роботу з Jupyter Notebooks, встановивши дистрибутив Anaconda.



Anaconda є найбільш широко використовуваним дистрибутивом Python для роботи з даними і поставляється з попередньо встановленими найбільш популярними бібліотеками та інструментами. Деякі з найбільших бібліотек Python, включених в Anaconda, включають NumPy (<https://numpy.org/>), pandas (<https://pandas.pydata.org/>) і Matplotlib (<https://matplotlib.org/>).

Повний список - <https://docs.anaconda.com/anaconda/packages/pkg-docs/>

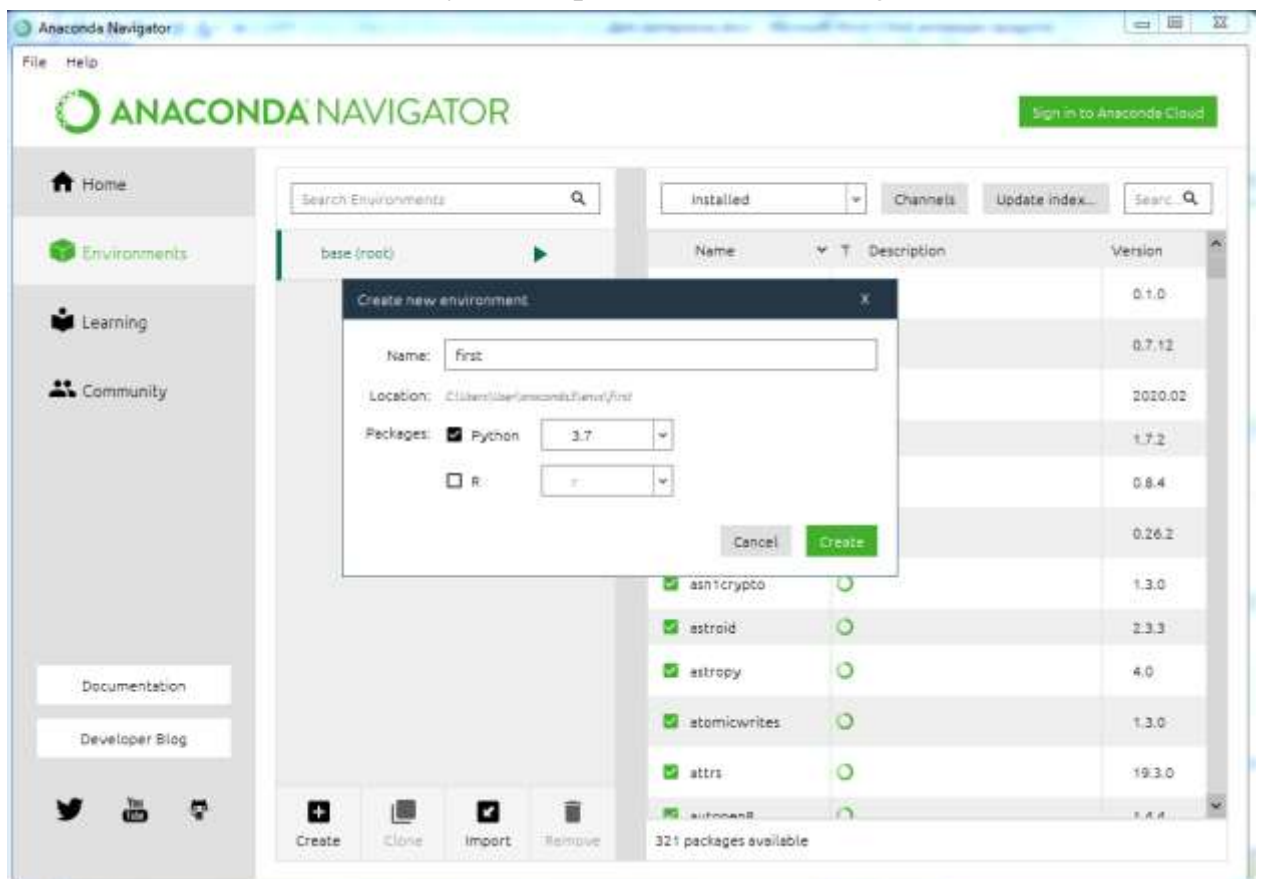
Це дозволяє приступити до роботи, без зайвого управління установками або занепокоєння про залежності і проблемах установки, пов'язаних з ОС.

Для встановлення Anaconda:

1. Завантажте останню версію Anaconda
<https://www.anaconda.com/products/individual>

2. Встановіть Anaconda виконуючи інструкціям.

Після встановлення буде відкрито anaconda navigator:



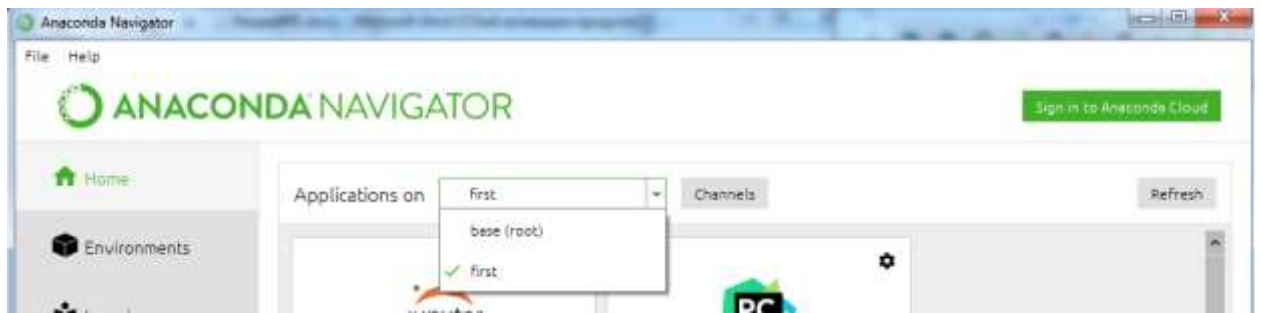
Для зручності роботи рекомендується створити окреме оточення - це дозволить уникнути конфліктів версій (наприклад коли потрібна інша версія пітона або бібліотеки). Для цього в терміналі потрібно запустити наступне:

```
conda create -n first python = 3.6
```

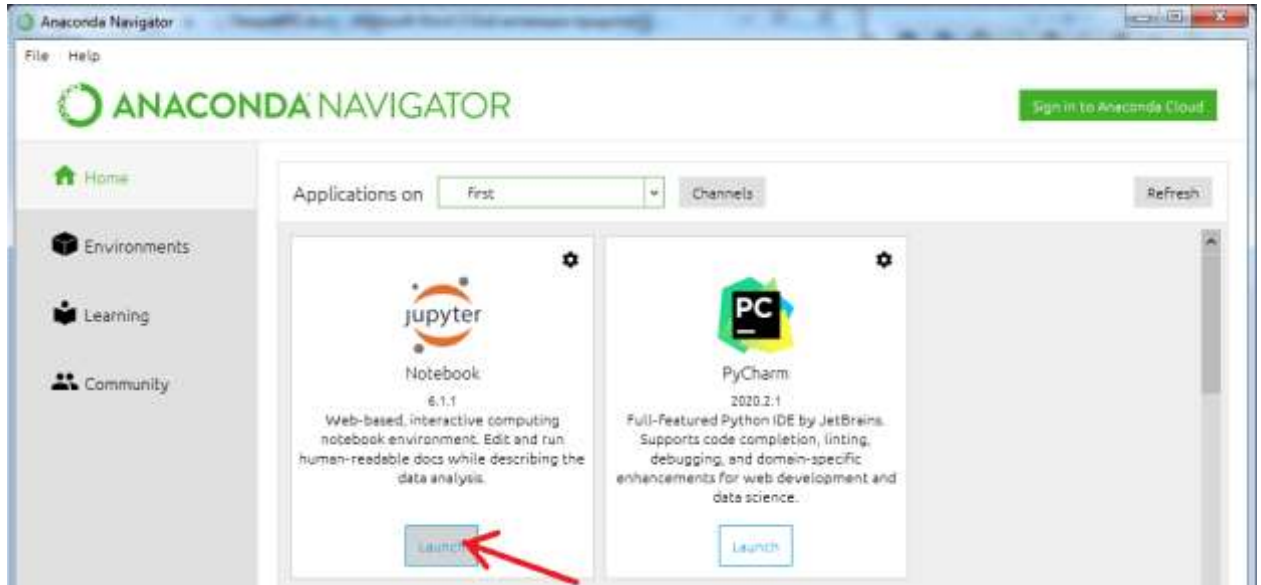
де **first** - назва оточення. Можна також вказати потрібну версію Python.

Створити нове оточення можна також і в вкладці **environments**, натиснувши на **Create**.

Створення оточення може зайняти кілька хвилин. Після установки в anaconda navigator можна вибрати оточення:



Далі потрібно встановити і запустити в обраному оточенні Jupyter Notebook.



Для просунутих користувачів

1. Встановлення Python – скачати файл завантаження з сайту <https://www.python.org/>

2. Запустити файл. Встановити PATH - Панель управління\Система и безопасность\Система – Дополнительные параметры системы, складка дополнительно

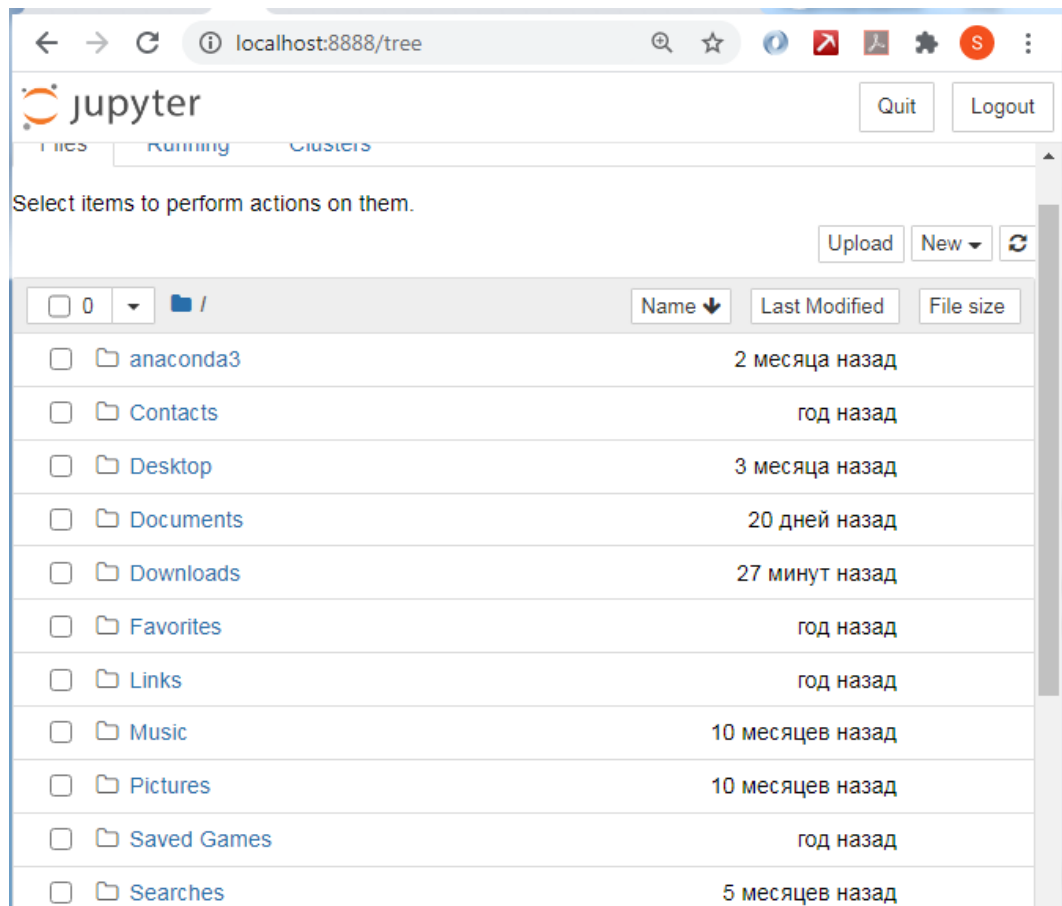
3. `pip3 install jupyter`

4. запуск блокнота

`jupyter notebook`

Створення першого блокнота

Після запуску блокнота в браузері відкриється нова вкладка, що буде виглядати наступним чином:

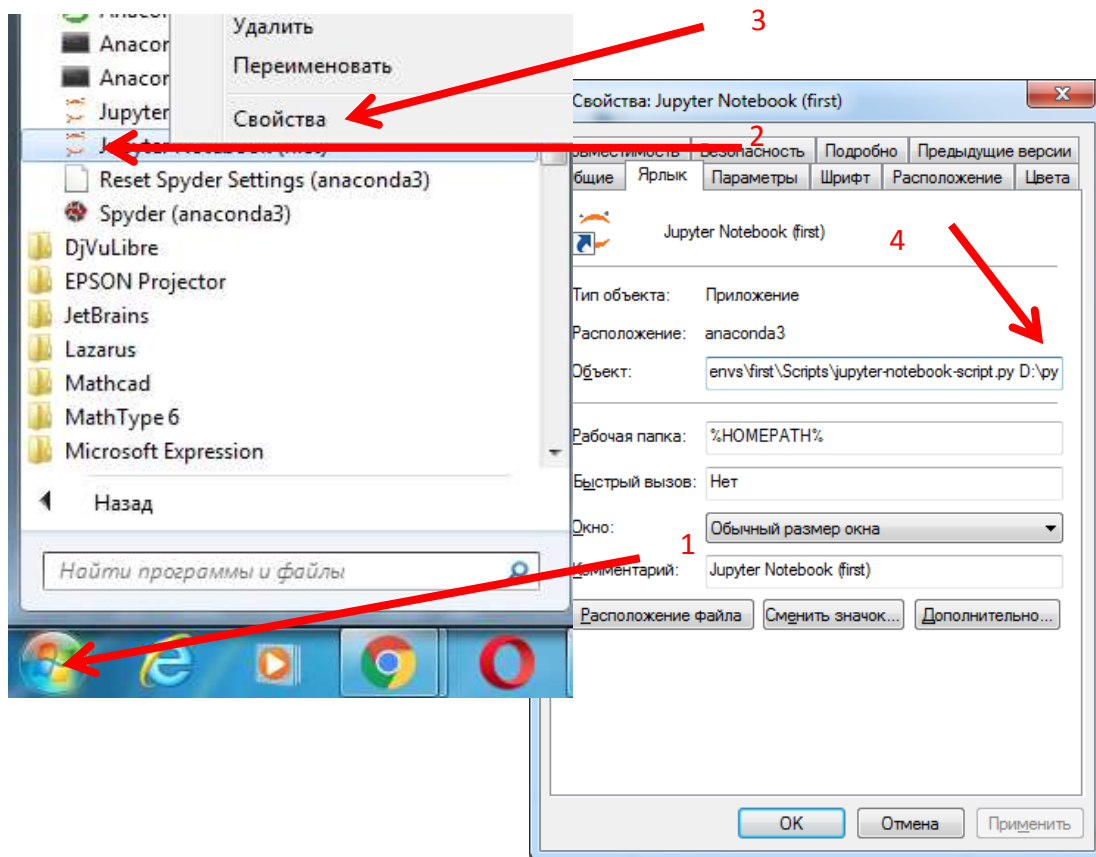


Це – панель інструментів ноутбука, створена для управління ноутбуками Jupyter.

Майте на увазі, що панель управління надасть вам доступ тільки до файлів і підпапках, що містяться в каталозі запуску Jupyter; проте каталог запуску може бути змінений.

При запуску із панелі Пуск:

1. Клацнути мишею Пуск>Все програми>Anaconda
2. Правою кнопкою миші клацнути на назві потрібного блокноту та обрати пункт «Свойства»
3. Замінити в полі «Объект» %USERPROFILE% на шлях до потрібної папки, Наприклад, D:\ру



При запуску з консолі

1. Перейти в потрібну папку, використовуючи команду `cd`.
2. Визвати команду `jupyter notebook`

```

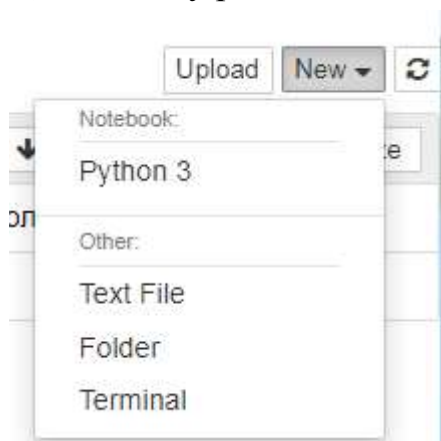
C:\Windows\system32\cmd.exe - jupyter notebook
Microsoft Windows [Version 6.1.7601]
(c) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.
C:\Users\User>cd /d d:\py
d:\py>jupyter notebook
[I 19:23:34.385 NotebookApp] The port 8888 is already in use, trying another port.
[I 19:23:34.387 NotebookApp] The port 8889 is already in use, trying another port.
[I 19:23:34.388 NotebookApp] The port 8890 is already in use, trying another port.
[I 19:23:34.389 NotebookApp] The port 8891 is already in use, trying another port.
[I 19:23:34.391 NotebookApp] The port 8892 is already in use, trying another port.
[I 19:23:34.392 NotebookApp] The port 8946 is already in use, trying another port.
[W 19:23:34.406 NotebookApp] Terminals not available (error was No module named
'winpty.cuwinpty')
  
```

Інші способи можна переглянути за посиланням <https://stackoverflow.com/questions/35254852/how-to-change-the-jupyter-start-up-folder>

URL-адреса в адресному рядку схожа на `http://localhost:8888/tree`. Localhost не є веб-сайтом, але вказує, що контент обслуговується з вашого локального комп'ютера: з вашого власного комп'ютера. Ноутбуки і панель інструментів Jupyter - це веб-додатки, і Jupyter запускає локальний веб сервер Python для обслуговування цих додатків в веб-браузері, що робить його

практично незалежним від платформи і відкриває можливості для більш зручного обміну в Інтернеті.

Для початку роботи натисніть на кнопку New і оберіть Python3

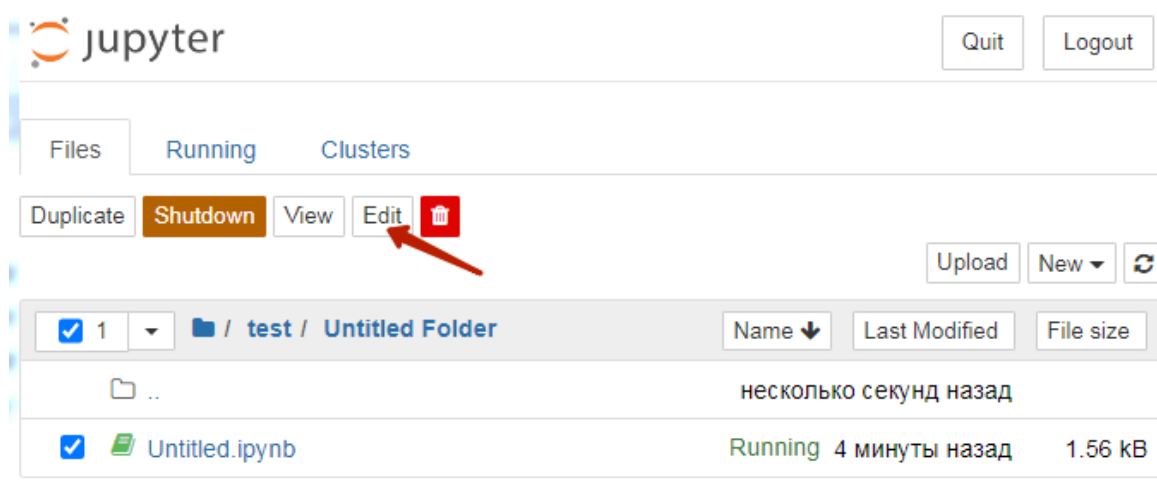


Ваш перший блокнот Jupyter відкриється в новій вкладці - кожен блокнот використовує свою вкладку, тому ви можете відкривати кілька блокнотів одночасно. Якщо ви перейдіть назад на панель інструментів, ви побачите новий файл `Untitled.ipynb` і зелений колір піктограми і тексту **Running**, який говорить про те, що ваш ноутбук працює.

Файл `.ipynb`

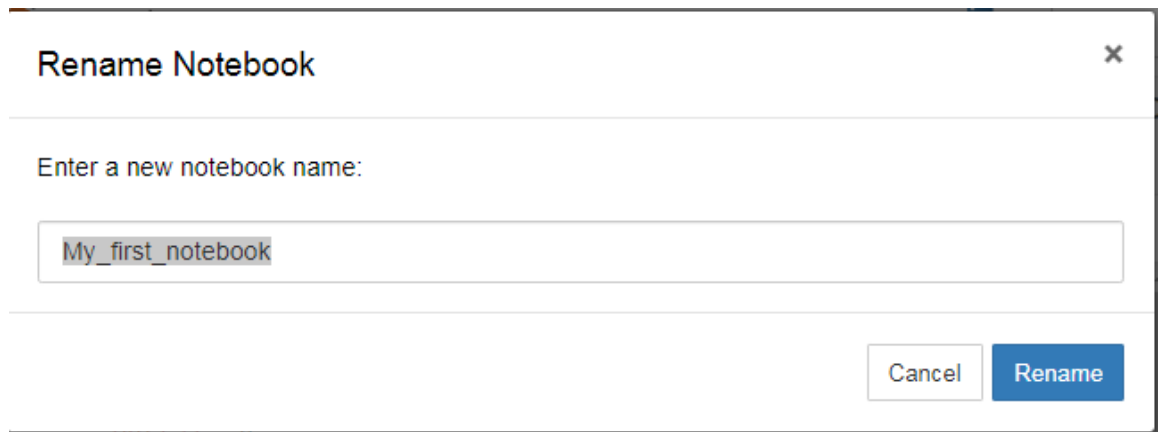
Буде корисно зрозуміти, що це за файл насправді. Кожен файл `.ipynb` є текстовим файлом, який описує вміст вашої записної книжки у форматі JSON. Кожна ланка й її вміст, включаючи вкладення зображень, які були перетворені в рядки тексту, перераховані в ньому разом з деякими метаданими. Ви можете редагувати їх самостійно - якщо знаєте, що робите! - вибравши «**Edit**» **Edit Notebook Metadata** в рядку меню в записнику.

Можна подивитися текст файлу в панелі навігації, натиснувши на кнопку **Edit**.



Перейменування файлу

Двічі клацнувши на назві файлу можна її змінити.



Ядра (kernels) та комірки (cells)

- kernel (ядро) - це «обчислювальний движок», який виконує код, що міститься в документі ноутбука.
- cell (комірка) - це контейнер для тексту, який буде відображатися в записнику, або код, який буде виконуватися ядром ноутбука.

Комірка (Cell)

Осередки утворюють структуру ноутбука. На скріншоті нового блокнота в наведеному вище розділі це поле з зеленим контуром. Є два основних типи осередків, які ми розглянемо:

- Комірка коду (Code) містить код, який повинен бути виконаний в ядрі, і відображає його виведення нижче.
- Комірка Markdown містить текст, відформатований з використанням Markdown, і відображає його висновок на місці під час запуску.

Коли ви запустите комірку коду, її висновок буде відображено нижче, а мітка зліва зміниться з `In []` на `In [1]`. Виведення осередку коду також є частиною документа, тому ви можете побачити його в цій статті. Ви завжди можете визначити різницю між кодом і осередками Markdown, тому що осередки коду мають цю мітку зліва, а осередки Markdown – не мають.

Наприклад ← немає мітки

In [4]: `import numpy as np`
`import matplotlib.pyplot as plt`
← є мітка

Частина «In» мітки просто коротка запис слова «Input», а номер мітки вказує, коли осередок була виконана в ядрі - в нашому випадку осередок була виконана першої. Запустіть осередок знову, і мітка зміниться на In [2], тому що тепер осередок була другою, запущеної в ядрі. Пізніше стане зрозуміліше, чому це так корисно, коли ми ближче познайомимося з ядрами.

Введемо наступний код в комірку:

```
In [*]: import time
        time.sleep (3) # сон у 3 секунди
```

Ця комірка не виводить нічого, але для її виконання потрібно три секунди. При цьому Jupyter показує, що осередок в даний момент працює, змінивши мітку на In [*].

Як правило, дані у вихідні комірки надходять з будь-яких текстових даних, спеціально надрукованих під час виконання осередків, а також з значення останнього рядка в комірці, будь то змінна, виклик функції або щось ще. Наприклад:

```
In [9]: def say_hello(name):
        return 'Hello, {}'.format(name)
        say_hello('Vasya')
```

```
Out[9]: 'Hello, Vasya!'
```

Режими ноутбука

Блокноти Jupyter мають два різні режими EDIT і COMMAND.

Режим редагування:

Щоб увійти в режим EDIT, натисніть клавішу Enter на клавіатурі або клацніть клітинку. Режим редагування можна визначити за зеленою рамкою навколо комірки із зеленим лівим полем. Коли ви перебуваєте в режимі редагування, ви можете вводити текст в клітинки.

```
In [ ]:
```

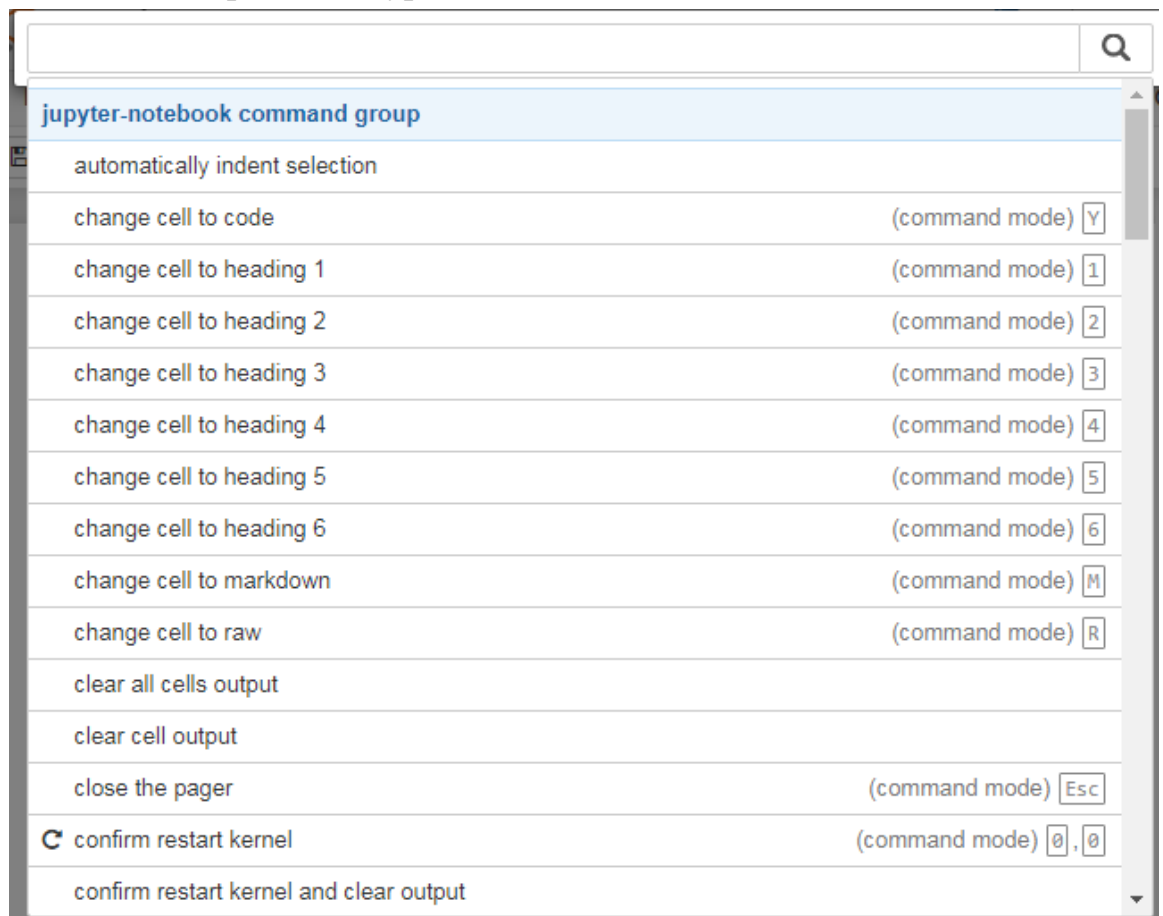
Командний режим:

Щоб увійти в режим COMMAND, натисніть Esc або клацніть де-небудь за межами комірки. Ви побачите сіру межу навколо комірки із синім лівим краєм. Коли ви перебуваєте в командному режимі, ви можете редагувати свій блокнот, але не можете вводити клітинки.

```
In [ ]:
```

Палітра команд

Натиснувши піктограму з клавіатурою, або **Ctrl+Shift+P**, відкривається палітра клавіатурних команд.



Приклади команд

Запустити вибрану комірку: **Ctrl + Enter**

Запустити комірку та вставити нову комірку нижче: **Alt + Enter**

Запустити комірку та перейти до клітинки нижче: **Shift + Enter**

Вставте клітинку вгорі: **Esc+A**

Вставте комірку внизу: **Esc+B**

Вирізати виділені клітинки: **Esc+X**

Копіювати виділені комірки: **Esc+C**

Як поділитись блокнотом.

1. Завантажити блокнот, наприклад на GitHub

2а. Ввести в адресний рядок https://colab.research.google.com/path_to_your_file_on_GitHub або

2.6. Перейти на сайт nbviewer <https://nbviewer.jupyter.org/> і у відповідному полі ввести адресу репозиторію GitHub. У відкритому вікні репозиторію обрати необхідний файл.

Але nbviewer не виконує код блокнотів, а тільки відображає вхід та вихід збережених ноутбуків як веб-сторінку.

Для виконання коду ноутбуку використовується сторонній веб-сервіс, що відкриває середовище виконання. Щоб завантажити ноутбук в mybinder.org необхідно натиснути кнопку *Execute on Binder* на панелі навігації.

