



3. Jupyter notebook 사용법



이번시간에 학습할 내용은..

python IDE들 중 Jupyter notebook의
사용법 및 활용학습

I. Jupyter notebook 기본 사용법

spyder ,pycham ,ipython 모두 IDE 툴 이쥬? 다 거기서 거기 아닌가요?

예 모두 파이썬 프로그램 코딩 및 실행을 하는데 유용한 도구 들이쥬. 각자 나름대로 특징들이 있겠쥬?

나름대로 특징이라면 프로그램 작성에서의 어떤 편리성을 말하는건가요?

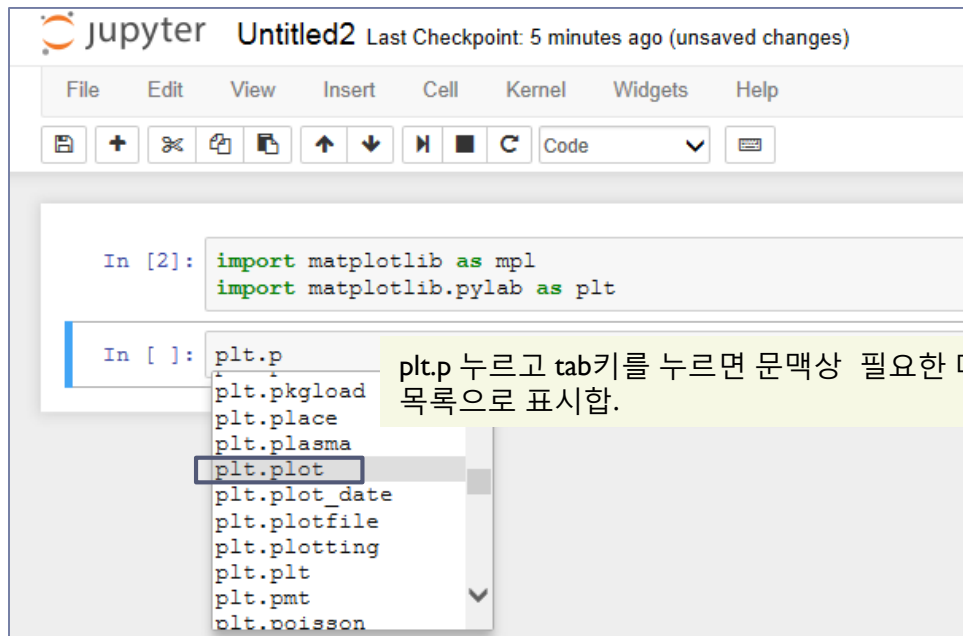
그렇쥬. 자동완성이라던가, 디버깅의 용이성이라던가 뭐 기타등등이쥬.

그럼 ipython 아니 이제는 Jupyter쥬? 이 툴도 어떤 유용한 기능들이 있나요?

예 사용자가 좋아할만한 여러 기능들을 갖고 있어요. 함께 익혀보쥬.

◆ 기본 기능

- 탭(tab) 키를 누르면 프로그램 흐름상 입력한 내용에 맞춰 적합한 변수명, 함수, 객체등을 표시
- 변수명 앞이나 뒤에 "?"기호를 붙이면 그 객체 에 대한 일반 정보를 출력



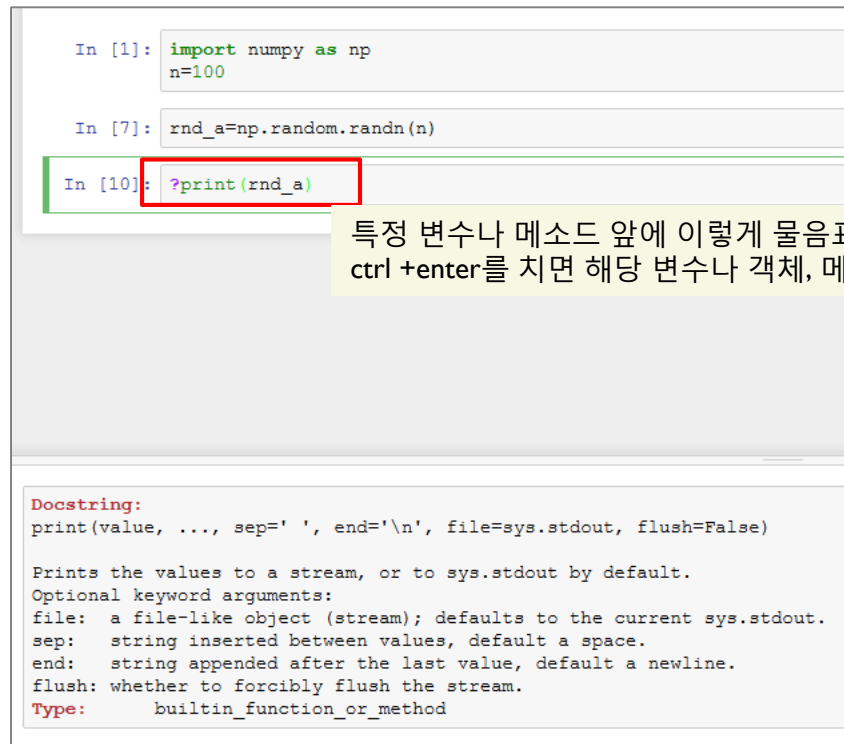
The screenshot shows the Jupyter Notebook interface. The top bar indicates 'Untitled2' and 'Last Checkpoint: 5 minutes ago (unsaved changes)'. The menu bar includes File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. Below the menu is a toolbar with icons for saving, adding cells, deleting, and running code. The main area contains a code cell with the following text:

```
In [2]: import matplotlib as mpl
import matplotlib.pyplot as plt

In [ ]: plt.p
```

A dropdown menu is open below the second line of code, showing a list of suggestions: plt.p, plt.pkgload, plt.place, plt.plasma, plt.plot, plt.plot_date, plt.plotfile, plt.plotting, plt.plt, plt.pmt, and plt.noisson. The 'plt.plot' option is highlighted.

plt.p 누르고 tab키를 누르면 문맥상 필요한 메소드들을 목록으로 표시함.



The screenshot shows the Jupyter Notebook interface. The top bar indicates 'Untitled2' and 'Last Checkpoint: 5 minutes ago (unsaved changes)'. The menu bar includes File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. Below the menu is a toolbar with icons for saving, adding cells, deleting, and running code. The main area contains a code cell with the following text:

```
In [1]: import numpy as np
n=100

In [7]: rnd_a=np.random.randn(n)

In [10]: ?print(rnd_a)
```

A red box highlights the '?print(rnd_a)' line. Below the code cell, the docstring for the 'print' function is displayed:

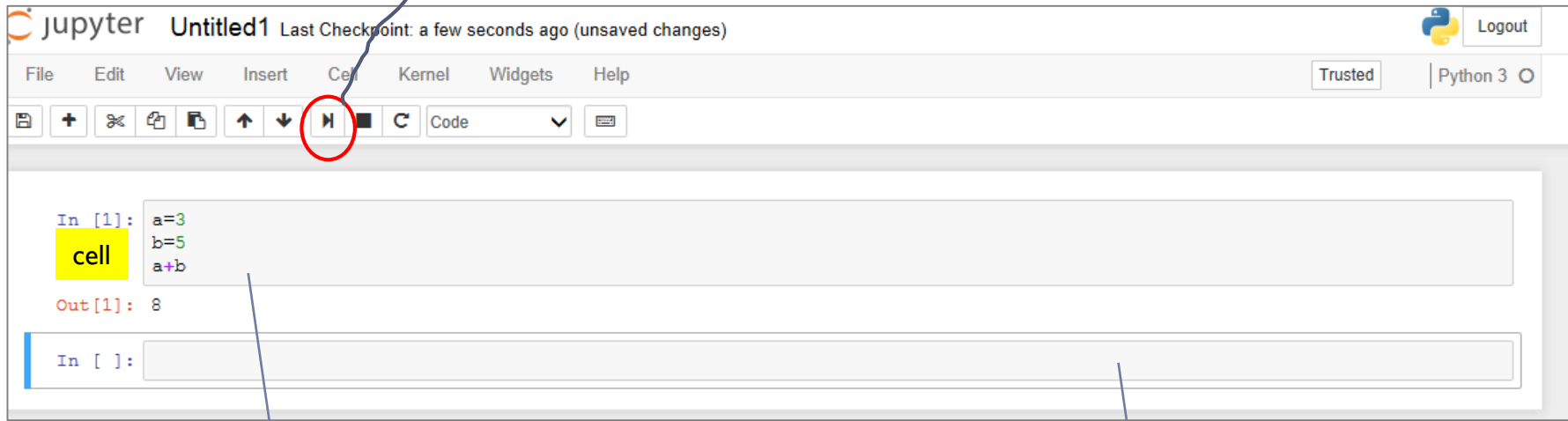
```
Docstring:
print(value, ..., sep=' ', end='\n', file=sys.stdout, flush=False)

Prints the values to a stream, or to sys.stdout by default.
Optional keyword arguments:
file:  a file-like object (stream); defaults to the current sys.stdout.
sep:   string inserted between values, default a space.
end:   string appended after the last value, default a newline.
flush: whether to forcibly flush the stream.
Type:  builtin_function_or_method
```

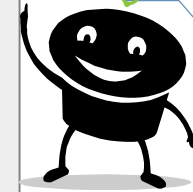
특정 변수나 메소드 앞에 이렇게 물음표(?)를 붙이고 ctrl +enter를 치면 해당 변수나 객체, 메소드에 대한 정보를 출력

◆ jupyter에서 코드를 실행하는 방법

1. run 단추를 눌러서 코드를 실행해요. → 현재 셀 실행



jupyter notebook 에서 모든 작업은 노트북 단위로 이뤄져요. 파일을 생성한다는건 노트북을 생성한다는 이야기에요.



2. ctrl + enter : 현재 셀을 실행

3. shift + enter : 현재 셀 실행하고 이렇게 다음 셀을 생성해요.



코드를 완성한 후 한꺼번에 실행하는 방법은 없나요?
난 그게 좋은데ㅠㅠ



아하..걱정말아요. 파일을 생성할 때 확장자를 .py형식으로 지정하면 돼요. 우리는 이런형식의 코드를 파일을 스크립트(script)라고해요.

Jupyter interface showing the file browser. The 'New' button is highlighted with a red box and labeled '1. 클릭'. The 'Text File' option in the dropdown menu is highlighted with a red box and labeled '2. 클릭'.

Jupyter editor window titled 'untitled.txt' showing a Python script:

```
1 import matplotlib as mpl
2 import matplotlib.pyplot as plt
3 plt.plot([2,5,7,12])
4 plt.show()
5
```

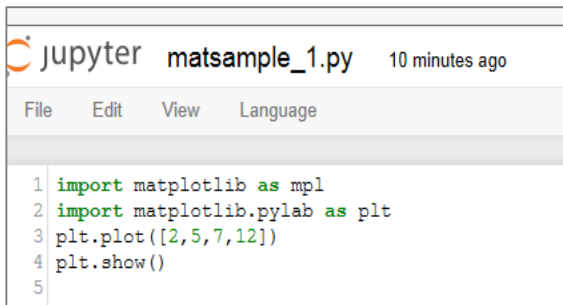
3. 이렇게 에디터 창이 나타나면 코딩을 해요.

4. notebook 이름변경위해 클릭

Jupyter interface showing the 'Rename File' dialog box. The 'untitled.txt' file is selected. The 'Enter a new filename:' field contains 'matsample_1.py'. The 'OK' button is highlighted with a red box and labeled '6. 클릭'.

5. 확장자를 .py로 입력해요.

%run 명령어 이용하여 실행하기



```
1 import matplotlib as mpl
2 import matplotlib.pyplot as plt
3 plt.plot([2,5,7,12])
4 plt.show()
5
```

리눅스에서는 terminal mode로 전환하여 손쉽게 .py 파일을 실행할 수 있어요.

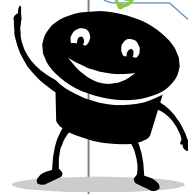
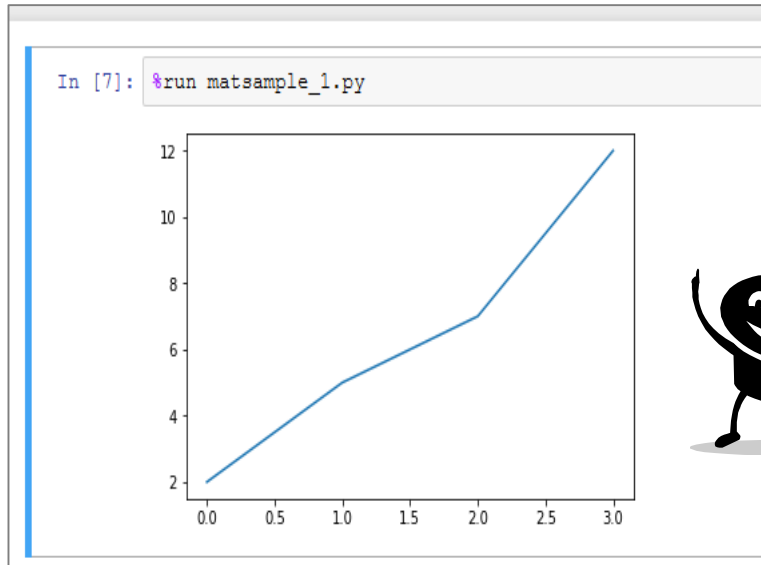
실행방법 : `python matsample_1.py`

현재 윈도우에서는 터미널 모드를 사용할 수 없어요. 따라서, 우리는 %run 명령어를 이용하여 jupyter 셀 안에서 파이썬 프로그램을 실행하는 방법을 알아보죠.

how to) 1. [new]-[python3]를 선택해요.



how to) 2. 아래 그림처럼 %run 명령을 입력하고 실행할 파이썬 파일명을 입력해요. 그리고, ctrl+enter를 치면 그림처럼 실행결과를 확인할 수 있어요.



%run 같은 명령어를 매직명령어
라고 해요. 매직명령어 궁금하죠?

2. 매직명령어 알아보기

- 매직명령어 - 작업의 유연성을 높이거나, 시스템 동작을 손쉽게 제어할 수 있는 특수한 명령어를 의미
- 형식 : %명령어

명령어	기능
%cd	디렉토리 경로 변경시 사용
%ls	파일이나 폴더 목록표시
%paste	클립보드에서 들여쓰기된 상태로 파이썬 코드 붙여넣기
%magic	도움말 출력
%quickref	빠른 도움말 표시
%hist	명령어 히스토리 출력
%run	Jupyter에서 파이썬 스크립트 실행
%time	명령 실행시간 출력

In [1]: %ls *.py

스크립트 파일목록을 표시

c 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: E028-E163

C:\Users\Administrator\Documents\aa 디렉터리

```
2014-11-19 오후 06:25          1,035 25-3.py
2014-11-20 오후 04:51           266 29-2.py
2017-10-15 오전 11:35           88 matsample_1.py
              3개 파일          1,389 바이트
              0개 디렉터리 56,075,927,552 바이트 남음
```

In [4]: %hist

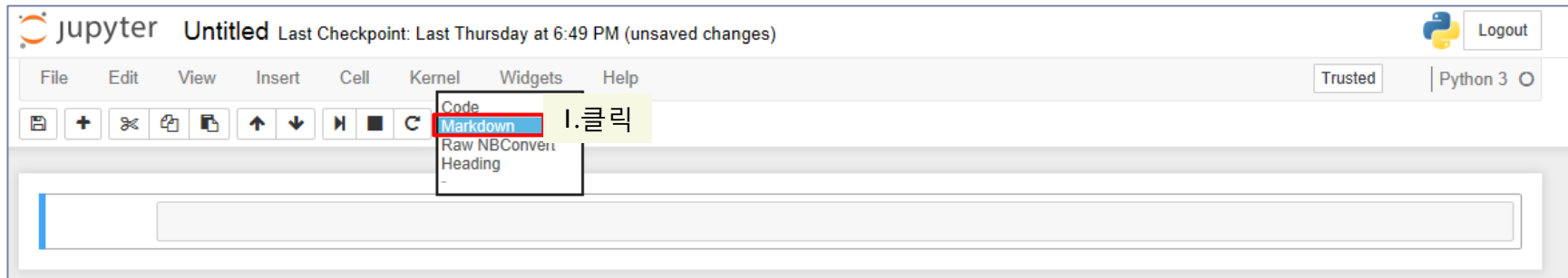
명령어 히스토리정보 표시

```
%ls
%run matsample_1.py
%logstart
%hist
```

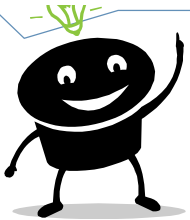

3. markdown 기능 익히기

➤ markdown - notebook을 텍스트 문서일부로 편집하거나 해당 코딩에 대한 근거자료를 추가하여 보고서 문서로도 활용

how to) 1. 아래 그림처럼 'code'목록에서 'markdown'을 선택한 후 제목을 입력해보아요.



markdown에서 헤드라인 입력할 때
글의 headline사이즈는 #으로
표시해요.



가장 큰 글씨

조금 작은 글씨

조금조금 작은 글씨

html 작성시 <h1>,<h2> <h3>..
과 비슷한거네!!



jupyter Untitled Last Checkpoint: Last Thursday at 6:49 PM (autosaved)

File Edit View Insert Cell Kernel Widgets Help

File Edit View Insert Cell Kernel Widgets Help

1. 클릭

2. 이렇게 입력을 하고 ctrl +enter !!

서울 지하철 평균 이용객 수 분석

가장 이용객 수 가 많은 지하철은 몇 호선 무슨 역일까요?

3. 이렇게 결과가 나타나요.

서울 지하철 평균 이용객 수 분석

가장 이용객 수 가 많은 지하철은 몇 호선 무슨 역일까요?

how to) 2. + 단추를 눌러서 새로운 셀에 pandas library 임포트하는 코딩을 해요.

File Edit View Insert Cell Kernel Widgets Help

1. 클릭

2. 코딩하고 ctrl+enter!!

In [2]: import pandas

how to) 3. + 단추를 누른 후 markdown으로 변경한 다음 그림처럼 글을 입력해보아요.

File Edit View Insert Cell Kernel Widgets Help

1. 클릭

2. 선택

3. 내용입력 후 ctrl +enter

서울 지하철 평균 이용객 수 분석

가장 이용객 수 가 많은 지하철은 몇 호선 무슨 역일까요?

In [2]: import pandas

데이터 science를 위한 가장 대표적인 library pandas를 이용해요.

jupyter Untitled Last Checkpoint: Last Thursday at 6:49 PM (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help

4. 위의 제목보다 조금 작게 표시되죠?

서울 지하철 평균 이용객 수 분석

가장 이용객 수 가 많은 지하철은 몇 호선 무슨 역일까요?

In [2]: import pandas

데이터 science를 위한 가장 대표적인 library pandas를 이용해요.

how to) 4. + 단추를 누른 후 csv파일을 읽어오는 동작을 코딩하고 ctrl+enter를 쳐서 실행해보아요.

서울 지하철 평균 이용객 수 분석

가장 이용객 수 가 많은 지하철은 몇 호선 무슨 역일까요?

```
In [2]: import pandas
```

데이터 science를 위한 가장 대표적인 library pandas를 이용해요.

```
In [3]: pl=pandas.read_csv('subway_data1.csv',encoding='utf-8')
```

1. 이렇게 코딩하고 ctrl+enter!!

how to) 5. + 단추를 눌러서 새로운 셀을 추가한 후 markdown으로 변경한 다음 그림처럼 글과 이미지를 입력해보아요.

서울 지하철 평균 이용객 수 분석

가장 이용객 수 가 많은 지하철은 몇 호선 무슨 역일까요?

```
In [2]: import pandas
```

데이터 science를 위한 가장 대표적인 library pandas를 이용해요.

```
In [3]: pl=pandas.read_csv('subway_data1.csv',encoding='utf-8')
```

read_csv를 입력한 후 shift+tab을 눌러보세요. 함수사용규칙을 상세히 알 수 있어요.

서울의 지하철 노선도를 참고로 보죠



데이터 science를 위한 가장 대표적인 library pandas를 이용해요.

```
In [3]: pl=pandas.read_csv('subway_data1.csv',encoding='utf-8')
```

read_csv를 입력한 후 shift+tab을 눌러보세요. 함수사용규칙을 상세히 알 수 있어요.

서울의 지하철 노선도를 참고로 보죠

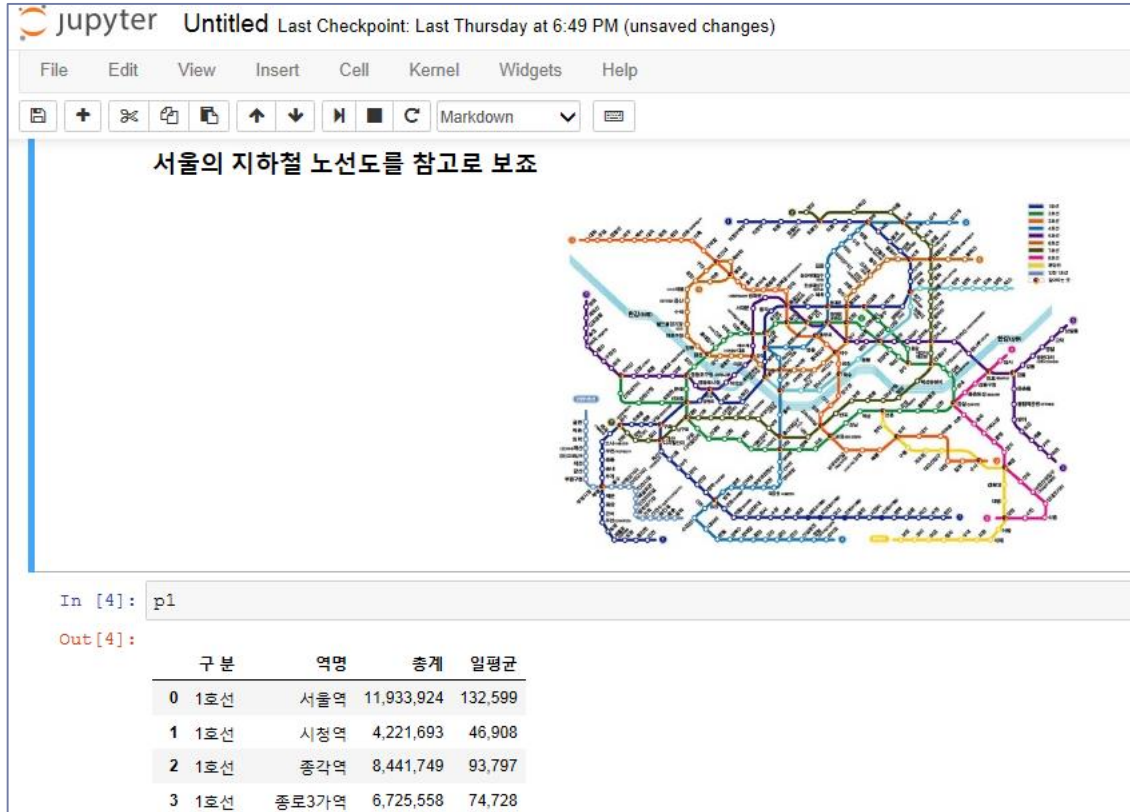
```

```

이미지 삽입하는 방법은 html과 똑같네요?



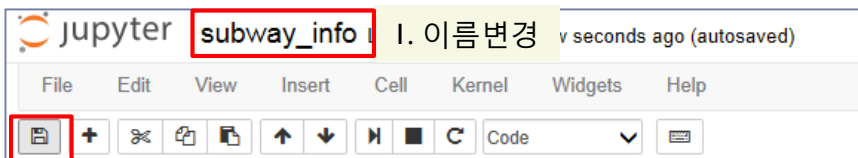
how to) 6. + 단추를 누른 후 입력된 결과를 화면에 표시하는 작업을 그림처럼 코딩해서 마무리해요.



이외에도
기울임꼴 *글자*
굵게 **글자**
굵게+기울임꼴 ***글자*** 와 같이
강조를 할 수 있어서 편리해요.

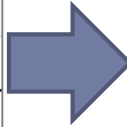
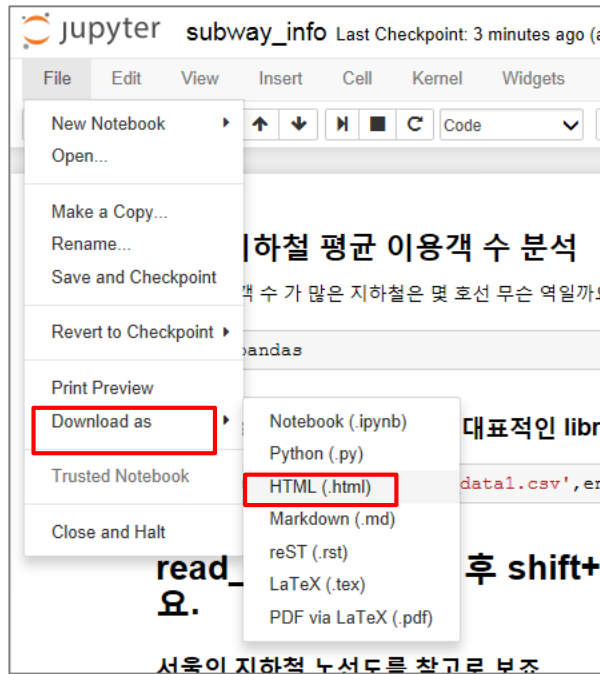


how to) 7. 작업한 노트북 이름을 “subway_info”로 변경한 후 저장해요.



2. 클릭

how to) 8. 작업한 노트북을 html 문서로 저장하여 웹브라우저에서 결과를 확인해보죠. [file]-[download as]-[html]을 선택해요.



The image shows a web browser window displaying the rendered HTML output of the Jupyter Notebook. The browser address bar shows the file path 'C:\Users\Administrator\Downloads\subway_info (1).html'. The rendered content includes a title '서울 지하철 평균 이용객 수 분석' (Seoul Subway Average Passenger Analysis), a question '가장 이용객 수가 많은 지하철은 몇 호선 무슨 역일까요?' (Which subway line and station have the most passengers?), and an introduction to the pandas library. It also includes a code cell for reading a CSV file and a table showing the average number of passengers for various subway lines and stations.

서울 지하철 평균 이용객 수 분석

가장 이용객 수 가 많은 지하철은 몇 호선 무슨 역일까요?

In [2]: `import pandas`

데이터 science를 위한 가장 대표적인 library pandas를 이용해요.

In [3]: `p1=pandas.read_csv('subway_data1.csv',encoding='utf-8')`

read_csv를 입력한 후 shift+tab을 눌러보세요. 함수사용규칙을 상세히 알 수 있어요.

서울의 지하철 노선도를 참고로 보죠

In [4]: `p1`

Out [4]:

	구 분	역명	총계	일평균
0	1호선	서울역	11,933,924	132,599
1	1호선	시청역	4,221,693	46,908
2	1호선	종각역	8,441,749	93,797
3	1호선	종로3가역	6,725,558	74,728
4	1호선	종로5가역	5,038,305	55,981
5	1호선	동대문역	3,116,104	34,623
6	1호선	신설동역	2,992,487	33,250
7	1호선	제기동역	3,757,112	41,746
8	1호선	청량리역	5,555,303	61,726