# Guida all'installazione di Python e Jupyter Notebook

In questa guida ci occupiamo di come installare Python e Jupyter Notebook.

## Installazione di Python

Le modalità per l'installazione di Python variano a seconda del sistema operativo che si utilizza.

#### **Python per Linux**

Per installare Python su Linux è sufficiente aprire una shell (ad esempio, su Ubuntu questa cosa si può fare con la combinazione di tasti "Ctrl+Alt+t") e lanciare i seguenti comandi (dove vi verrà chiesto di confermare l'operazione con la vostra password):

```
sudo apt-get install python3
sudo apt-get install python-pip
sudo apt-get install idle3
```

#### Python per Windows e MacOS

Per quanto riguarda l'installazione di Python per i sistemi operativi Windows e MacOS, bisogna fare il download di un installer ed eseguirlo. In particolare, è sufficiente recarsi all'indirizzo <u>https://www.python.org/downloads/release/python-374/</u> e fare il download dell'installer (executable installer per Windows) a seconda del sistema operativo che si utilizza e della versione (se a 32 o a 64 bit). Durante l'installazione si consiglia di spuntare la casella "Add Python 3.7 to PATH" (questo vi verrà chiesto ad un certo punto dell'installazione).

## Installazione ed utlizzo di Jupyter Notebook

Per le seguenti installazioni abbiamo bisogno di aprire una shell (anche detto prompt, o console) dei comandi. Per aprire una shell sul sistema operativo Windows si può utilizzare la combinazione di tasti "Windows+R" poi digitare cmd ed infine premere invio. Invece per aprire una shell sul sistema operativo MacOS è sufficiente cercare l'applicazione *terminale* e lanciarla.

Installeremo Jupyter Notebook usando il package manager pip di Python. Per prima cosa aggiornate il package all'ultima sua versione, aprendo una shell ed eseguendo il comando

python3 -m pip install --upgrade pip

Per installare Jupyter Notebook adesso è sufficiente eseguire il seguente comando:

python3 -m pip install jupyter

Vediamo ora come utilizzare Jupyter Notebook. Per aprire ques'ultimo è sufficiente recarsi nella shell e lanciare il seguente comando:

jupyter notebook

All'avvio di Jupyter Notebook, il vostro browser aprirà la *notebook dashboard*, che funge da home page dell'applicazione. In particolare, dalla scheda *Files* potete navigare all'interno del vostro file system e selezionare i file notebook da aprire (questi file hanno estensione .ipynb). Ad esempio, la dashboard in Figura 2 mostra la pagina per la cartella AppuntiFondamenti1, che contiene, alcuni file ipynb.

AppuntiFondamenti/ × +	
→ C ③ localhost:8888/tree/AppuntiFondamenti	\$ 0
💭 jupyter	Quit Logout
Files Running Clusters	
Select items to perform actions on them.	Upload New -
0 V AppuntiFondamenti	Name      Last Modified File size
۵	alcuni secondi fa
🗆 🗐 If.ipynb	un giorno fa 27.9 kB
🗆 🕘 Intro.ipynb	un giorno fa 18.2 kB
Stringhe.ipynb	un giorno fa 20.3 kB
🗆 / Variabili.ipynb	un giorno fa 16.9 kB

[Figura 2]

Cliccando su un file, questo verrà aperto in una scheda del browser (cf. Figura 3)



[Figura 3]

Jupiter Notebook è un ambiente interattivo: è possibile infatti eseguire i frammenti di codice riportati nel file direttamente all'interno del browser stesso. La schermata in Figura 3, ad esempio, mostra il file Intro.ipynb, dopo l'esecuzione dell'istruzione Python 7\*(5+2), che ha prodotto il risultato 49 (cf. riga Out[1]: 49 ).

In effetti il file .ipynb che abbiamo aperto è un documento diviso in "celle" contenenti testo oppure codice Python. Le celle possono essere selezionate per essere modificate (editing mode) o per essere eseguite (command mode). La cella selezionata viene evidenziata da un riquadro con una sottile banda laterale di colore verde per l'editing mode, blu per il command mode.

Per selezionare una cella è sufficiente cliccare su di questa con il mouse. Se usate Jupiter Notebook solo per consultare le note distribuite in questo corso, o comunque non per produrre documenti, vi basta selezionare le celle contenenti codice. Ognuna di queste celle contiene un'area di editing, per editare i comandi (a destra della scritta In [], cf. Figura 3), ed un'area per visualizzare il risultato (a destra della scritta Out [], e visibile dopo aver eseguito il codice). Cliccando sull'area di editing avrete selezionato la cella in editing mode, e potrete modificare i comandi da eseguire. Cliccando invece sulle altre parti della cella avrete selezionato la cella in command mode, e potrete eseguire il codice in essa contenuto. Per eseguire il codice dovete selezionare il tasto Run (cf. Figura 3).

Se volete azzerare le precedenti esecuzioni e ripulire il documento dagli output prodotti, dal menu Kernel selezionate Restart & clear output. E' possibile anche solo riavviare il kernel (cioè riavviare l'interprete Python – il che comporta la perdita del contenuto corrente delle variabili usate).

Notiamo infine che per chiudere un file in esecuzione (ricordiamo che ogni file ha in effetti un terminale attivo – il "kernel"), non è sufficiente chiudere la scheda del browser che lo visualizza, ma bisogna accedere alla scheda "Running" della dashboard e selezionare Shutdown in corrispondenza del file che si intende chiudere (cf. Figura 5)

AppuntiFondamenti/ × +	
→ C ③ localhost:8888/tree/AppuntiFondamenti#running	☆ 💿
💭 Jupyter	Quit Logout
Files Running Clusters	
Currently running Jupyter processes	2
Terminals -	
There are no terminals running.	
Notebooks 🔻	
AppuntiFondamenti/Intro invnb	Python 3 Shutdown alcuni secondi fa

### [Figura 5]

Per altri dettagli su Jupiter Notebook ed una completa descrizione delle sue funzionalità si rimanda alla documentazione disponibile on-line (ad esempio <u>https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what\_is\_jupyter.html</u>)