

# Programmazione e Algoritmi 1

## Interazione con Python

prof. Gianluca Amato

Corso di Laurea in Economia e Informatica per l'Impresa  
a.a. 2023/24

2 ottobre 2023

## Interazione

Gianluca  
Amato

Il terminale e  
la shell

Interazione  
con Python  
tramite  
terminale

L'ambiente di  
sviluppo Visual  
Studio Code

Interazione  
con Python  
tramite Visual  
Studio Code

Obiettivi di queste slide:

- Conoscere le modalità di interazione con Python.
- Presentare l'ambiente di sviluppo Visual Studio Code.

Parti sul libro di testo corrispondenti:

- sezione 1.4
- suggerimenti per la programmazione 1.1

Il trattamento del libro di testo è comunque solo parziale.

Si suppone che lo studente abbia già installato Python, Visual Studio Code e tutte le estensioni correlate. In caso contrario, consultare la pagina “[Installazione dell'ambiente di sviluppo](#)” sul sito del corso.

## Interazione

Gianluca  
Amato

### Il terminale e la shell

Interazione  
con Python  
tramite  
terminale

L'ambiente di  
sviluppo Visual  
Studio Code

Interazione  
con Python  
tramite Visual  
Studio Code

- 1 Il terminale e la shell
- 2 Interazione con Python tramite terminale
- 3 L'ambiente di sviluppo Visual Studio Code
- 4 Interazione con Python tramite Visual Studio Code

Il **terminale** è un programma che permette di interagire con il computer tramite comandi testuali.

Questo metodo di interazione prende il nome di **interfaccia a riga di comando** o, in inglese, **Command Line Interface (CLI)**.

Tutti i sistemi operativi hanno un terminale, che cambia nome a seconda del caso:

- Windows 10: *Command Prompt* + *Windows PowerShell*
- Windows 11 come Windows 10 + *Windows Terminal*
- macOS: *Terminal*
- Linux (con interfaccia grafica GNOME): *Terminal*

Il terminale è solo il programma grafico che permette di interagire con il computer tramite comandi testuali. Il vero programma che esegue i comandi si chiama **shell**.

A parte l'aspetto grafico, i comandi supportati da una shell cambiano molto da un sistema operativo ad un altro.

Ad esempio, il comando per visualizzare il contenuto di una cartella è:

- *Command prompt*: `dir`
- *Windows PowerShell*: `dir` oppure `ls`
- *macOS*: `ls`
- *Linux*: `ls`

In generale:

- Linux e macOS hanno comandi simili;
- Windows PowerShell è abbastanza diversa ma ha alcuni comandi simili a Linux/macOS;
- Command Prompt è completamente diverso (e anche oltraggiosamente limitato).

## Interazione

Gianluca Amato

### Il terminale e la shell

Interazione con Python tramite terminale

L'ambiente di sviluppo Visual Studio Code

Interazione con Python tramite Visual Studio Code

```

amato@gianlu:~
[amato@gianlu ~]$ ls
ApacheDirectoryStudio  Dropbox      mnt          Postman      sitodec.zip  tmp
Desktop                git          Music        prova.py     src          Videos
Documents              gitcommand.txt Nextcloud    Public       svn          Zotero
Downloads              hello.py     Pictures     sitodec      Templates

[amato@gianlu ~]$ cat hello.py
print("Hello, world!")

[amato@gianlu ~]$ df
Filesystem      1K-blocks      Used Available Use% Mounted on
devtmpfs        4096            0      4096   0% /dev
tmpfs           16313640       26072  16287568 1% /dev/shm
tmpfs           6525456        2736   6522720  1% /run
/dev/nvme0n1p3 1951850496    249770688 1699702176 13% /
/dev/nvme0n1p3 1951850496    249770688 1699702176 13% /home
tmpfs           16313640       10880  16302760  1% /tmp
/dev/nvme0n1p2 996780         301364   626604  33% /boot
/dev/nvme0n1p1 613184         17780   595404  3% /boot/efi
tmpfs           3262728        3872   3258856  1% /run/user/1000
[amato@gianlu ~]$
  
```

Vediamo due comandi della shell:

- `ls`: il comando `ls` mostra il contenuto della **cartella corrente**. Inizialmente, la cartella corrente è la **home directory** dell'utente, ma la si può cambiare con il comando `cd`.
- `cd`: cambia la cartella corrente con quella specificata. Ad esempio, `cd Desktop` cambia la cartella corrente con la cartella `Desktop` (se esiste) che si trova all'interno della cartella corrente.

Questo è il risultato dell'utilizzo in sequenza dei comandi "`ls`", "`cd Desktop`" e di nuovo "`ls`".

```

amato@gianlu:~/Desktop
[amato@gianlu ~]$ ls
20tab.txt          Downloads          mnt               Postman          Templates
ApacheDirectoryStudio  Dropbox           Music             Public           tmp
Desktop            git               Nextcloud         src              Videos
Documents          gitcommand.txt   Pictures          svn              Zotero
[amato@gianlu ~]$ cd Desktop/
[amato@gianlu Desktop]$ ls
prova.py
[amato@gianlu Desktop]$
    
```

Una volta, prima dell'avvento delle interfacce grafiche, la shell era l'unico mezzo per comunicare con il sistema operativo.

La shell, per chi la sa usare, conserva ancora dei vantaggi rispetto all'interfaccia grafica:

- per la maggior parte dei compiti, è più veloce dare un comando con la shell che aprire finestre e finestrelle, anche perché non bisogna mai spostare le mani dalla tastiera per raggiungere il mouse;
- è programmabile: la shell è un vero e proprio linguaggio di programmazione con il quale è possibile automatizzare l'esecuzione di vari compiti;
- alcune attività di basso livello richiedono ancora l'uso della shell;
- è utilizzabile da un computer remoto, e in questo modo si ha l'accesso totale alla macchina, che può essere amministrata senza bisogno di essere presenti fisicamente nello stesso luogo.

## Interazione

Gianluca  
Amato

Il terminale e  
la shell

Interazione  
con Python  
tramite  
terminale

L'ambiente di  
sviluppo Visual  
Studio Code

Interazione  
con Python  
tramite Visual  
Studio Code

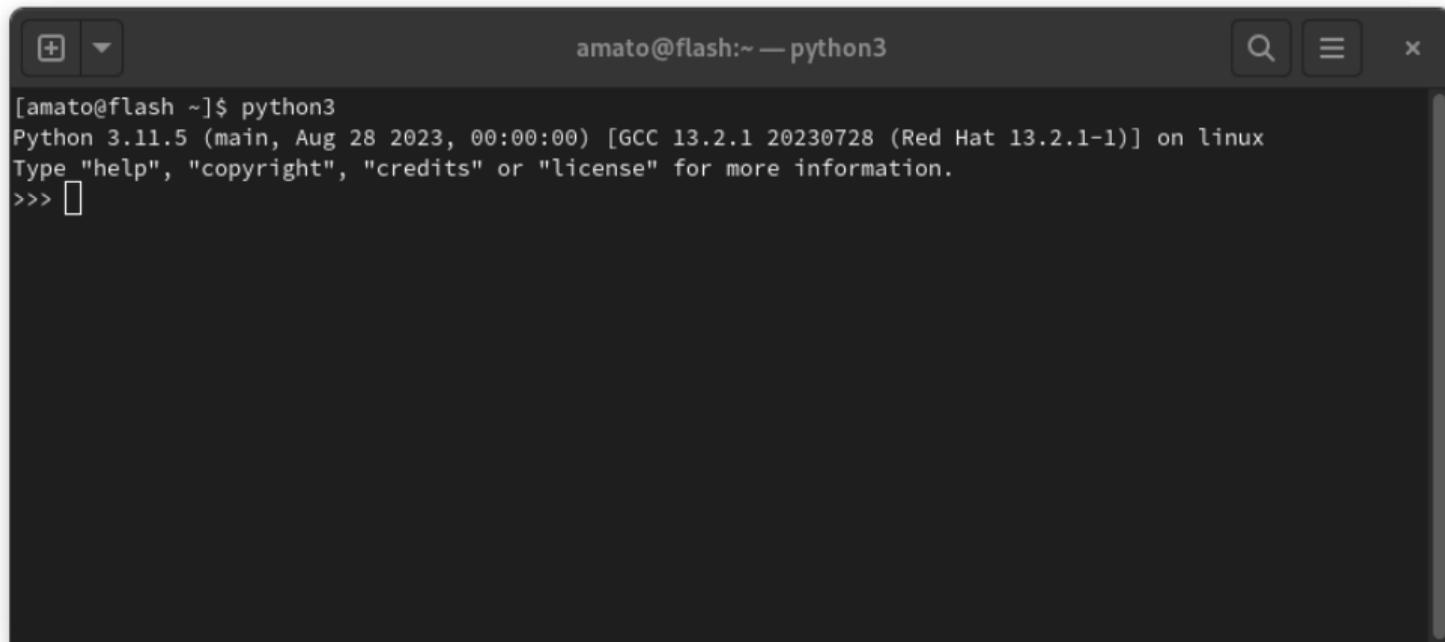
1 Il terminale e la shell

2 Interazione con Python tramite terminale

3 L'ambiente di sviluppo Visual Studio Code

4 Interazione con Python tramite Visual Studio Code

- Dal terminale, si lancia il comando `python3`.  
(in alcuni sistemi si può usare anche `python` al posto di `python3`).
- Compare il **prompt** di python, ovvero la stringa `>>>`



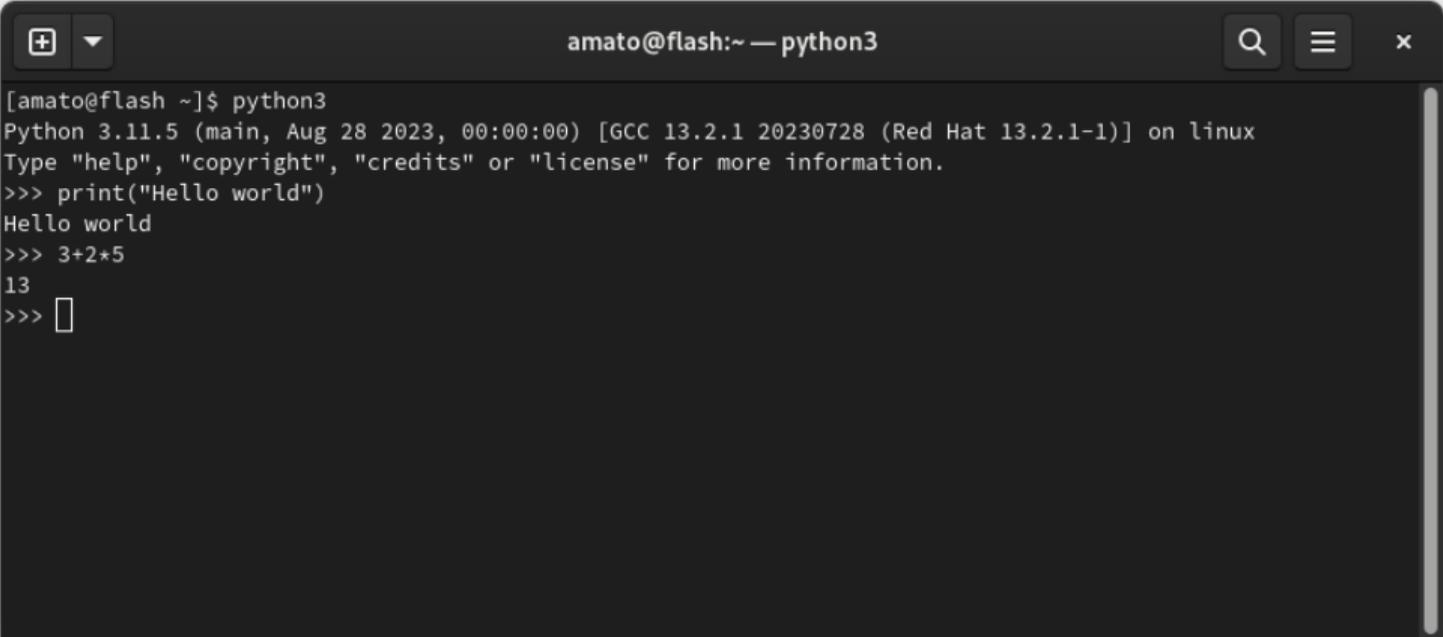
```

amato@flash:~ — python3
[amato@flash ~]$ python3
Python 3.11.5 (main, Aug 28 2023, 00:00:00) [GCC 13.2.1 20230728 (Red Hat 13.2.1-1)] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> █
    
```

## Uso interattivo dal terminale (2)

A questo punto si possono dare dei comandi che vengono eseguiti immediatamente, alla pressione del tasto `Enter` (ad esempio, è possibile usare Python come calcolatrice).

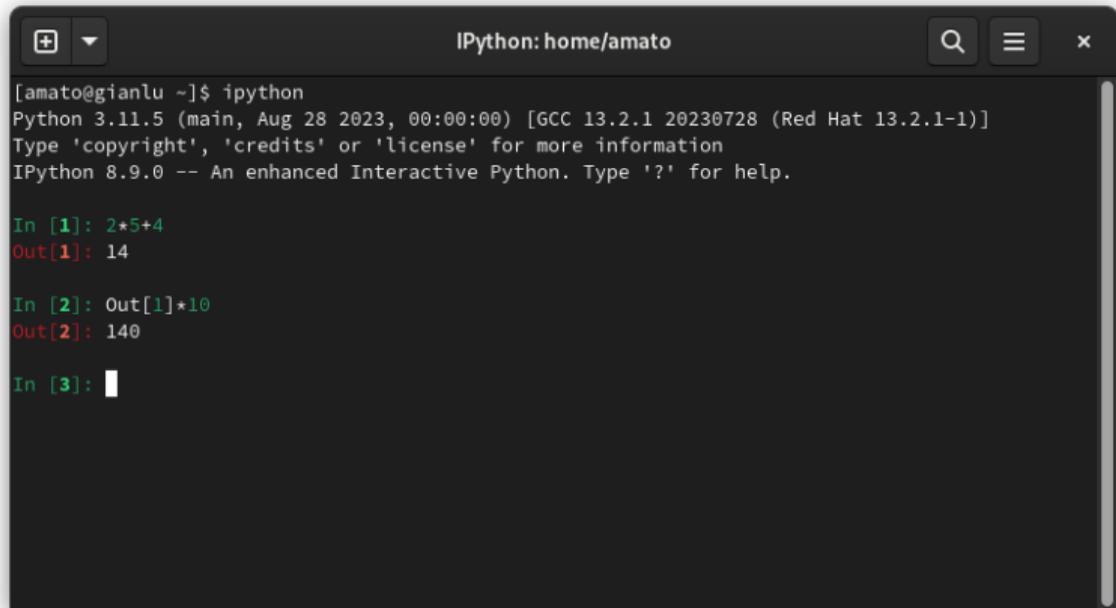
Per uscire si può dare il comando `quit()`.



```
amato@flash:~ — python3
[amato@flash ~]$ python3
Python 3.11.5 (main, Aug 28 2023, 00:00:00) [GCC 13.2.1 20230728 (Red Hat 13.2.1-1)] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello world")
Hello world
>>> 3+2*5
13
>>> █
```

Se disponibile, è preferibile usare il comando `ipython` al posto di `python`. Questo comando fornisce un'interfaccia interattiva più avanzata, con funzionalità aggiuntive.

Ad esempio, è possibile riferirsi all'output di un comando precedente con la sintassi `Out [n]`.



```

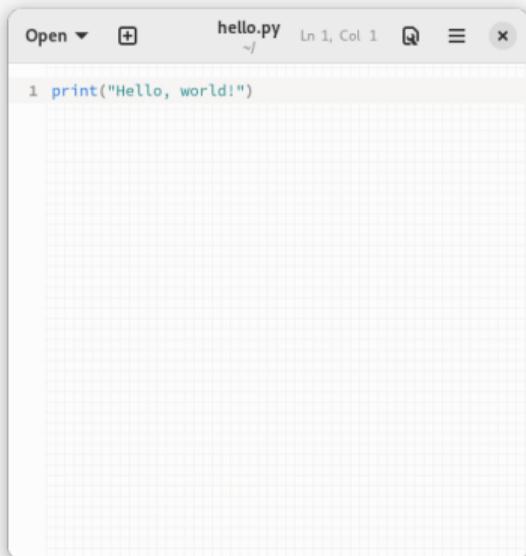
IPython: home/amato
[amato@gianlu ~]$ ipython
Python 3.11.5 (main, Aug 28 2023, 00:00:00) [GCC 13.2.1 20230728 (Red Hat 13.2.1-1)]
Type 'copyright', 'credits' or 'license' for more information
IPython 8.9.0 -- An enhanced Interactive Python. Type '?' for help.

In [1]: 2*5+4
Out[1]: 14

In [2]: Out[1]*10
Out[2]: 140

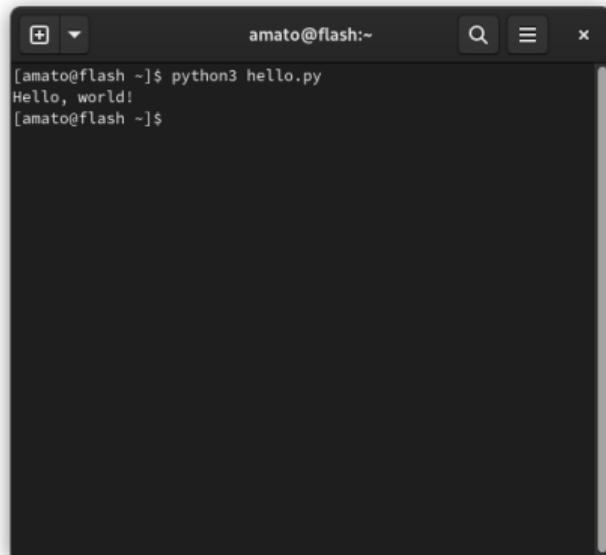
In [3]:
  
```

- Scrivere un file (generalmente con estensione `.py`) con un editor di testi.
- Aprire il terminale e utilizzare il comando `cd` per spostarsi nella cartella in cui si trova il file (se necessario).
- Eseguire il programma dal terminale con il comando `python3 nomefile.py`.



```

Open ▾ + hello.py Ln 1, Col 1
1 print("Hello, world!")
  
```



```

amato@flash:~
[amato@flash ~]$ python3 hello.py
Hello, world!
[amato@flash ~]$
  
```

## Interazione

Gianluca  
Amato

Il terminale e  
la shell

Interazione  
con Python  
tramite  
terminale

L'ambiente di  
sviluppo Visual  
Studio Code

Interazione  
con Python  
tramite Visual  
Studio Code

- 1 Il terminale e la shell
- 2 Interazione con Python tramite terminale
- 3 L'ambiente di sviluppo Visual Studio Code**
- 4 Interazione con Python tramite Visual Studio Code

## Interazione

Gianluca  
Amato

Il terminale e  
la shell

Interazione  
con Python  
tramite  
terminale

L'ambiente di  
sviluppo Visual  
Studio Code

Interazione  
con Python  
tramite Visual  
Studio Code

Un **IDE (Integrated Development Environment)** è un programma che consente di scrivere, modificare, compilare ed eseguire programmi. Inoltre, fornisce strumenti per il debug e per la gestione dei progetti.

Visual Studio Code è un IDE gratuito e open source:

- È disponibile per Windows, macOS e Linux.
- È molto leggero e veloce.
- È estendibile tramite plugin.
- È molto popolare tra gli sviluppatori.

Per poter utilizzare Python con Visual Studio Code è necessario installare l'apposita estensione.



## Interazione

Gianluca Amato

Il terminale e la shell

Interazione con Python tramite terminale

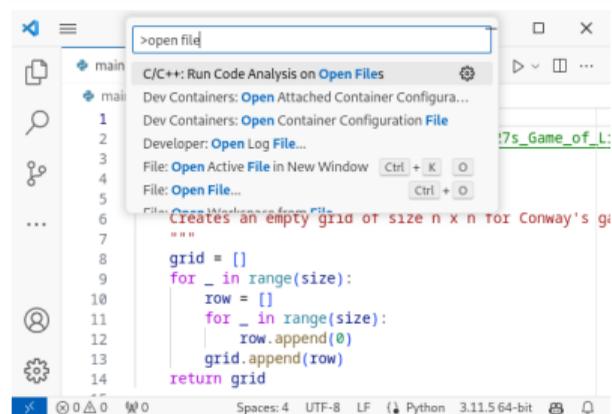
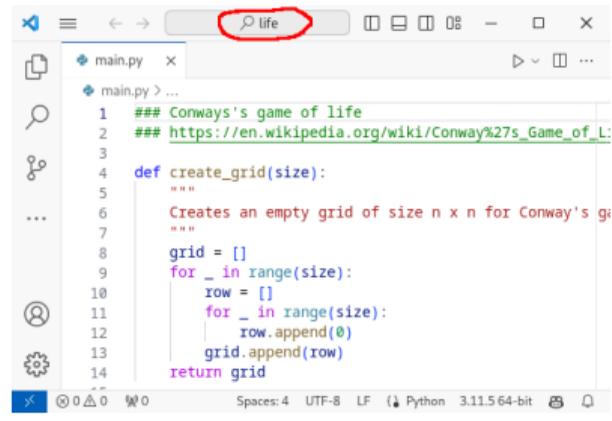
L'ambiente di sviluppo Visual Studio Code

Interazione con Python tramite Visual Studio Code

Non è nostro obiettivo presentare tutte le funzionalità di Visual Studio Code, lo imparerete usandolo (e potete consultare la [guida ufficiale](#), o altra documentazione on-line).

Una cosa fondamentale è però capire come dare comandi a Visual Studio Code. Per questo, è sufficiente:

- cliccare in alto nella finestra del software, in corrispondenza dell'icona della lente d'ingrandimento;
- premere il tasto ;
- iniziare digitare il comando che si vuole eseguire;
- selezionare il comando desiderato con il mouse (o selezionarlo con i tasti cursore e premere .



## Interazione

Gianluca Amato

Il terminale e la shell

Interazione con Python tramite terminale

L'ambiente di sviluppo Visual Studio Code

Interazione con Python tramite Visual Studio Code

Non è nostro obiettivo presentare tutte le funzionalità di Visual Studio Code, lo imparerete usandolo (e potete consultare la **guida ufficiale**, o altra documentazione on-line).

In alternativa, si può utilizzare la tastiera:

- premere la combinazione **Ctrl** + **Shift** + **P** su Linux/Windows o **Cmd** + **Shift** + **P** su macOS;
- iniziare a digitare il comando che si vuole eseguire (senza il simbolo **>**);
- selezionare il comando desiderato con i tasti cursore (o con il mouse) e premere **Enter**.

```

1  ### Conway's game of life
2  ### https://en.wikipedia.org/wiki/Conway%27s_Game_of_L
3
4  def create_grid(size):
5      """
6      Creates an empty grid of size n x n for Conway's gr
7      """
8      grid = []
9      for _ in range(size):
10         row = []
11         for _ in range(size):
12             row.append(0)
13         grid.append(row)
14     return grid

```

```

>open file
C/C++: Run Code Analysis on Open Files
Dev Containers: Open Attached Container Configura...
Dev Containers: Open Container Configuration File
Developer: Open Log File...
File: Open Active File in New Window  Ctrl + K  O
File: Open File...  Ctrl + O

```

## Interazione

Gianluca  
Amato

Il terminale e  
la shell

Interazione  
con Python  
tramite  
terminale

L'ambiente di  
sviluppo Visual  
Studio Code

Interazione  
con Python  
tramite Visual  
Studio Code

- 1 Il terminale e la shell
- 2 Interazione con Python tramite terminale
- 3 L'ambiente di sviluppo Visual Studio Code
- 4 Interazione con Python tramite Visual Studio Code

## Interazione

Gianluca Amato

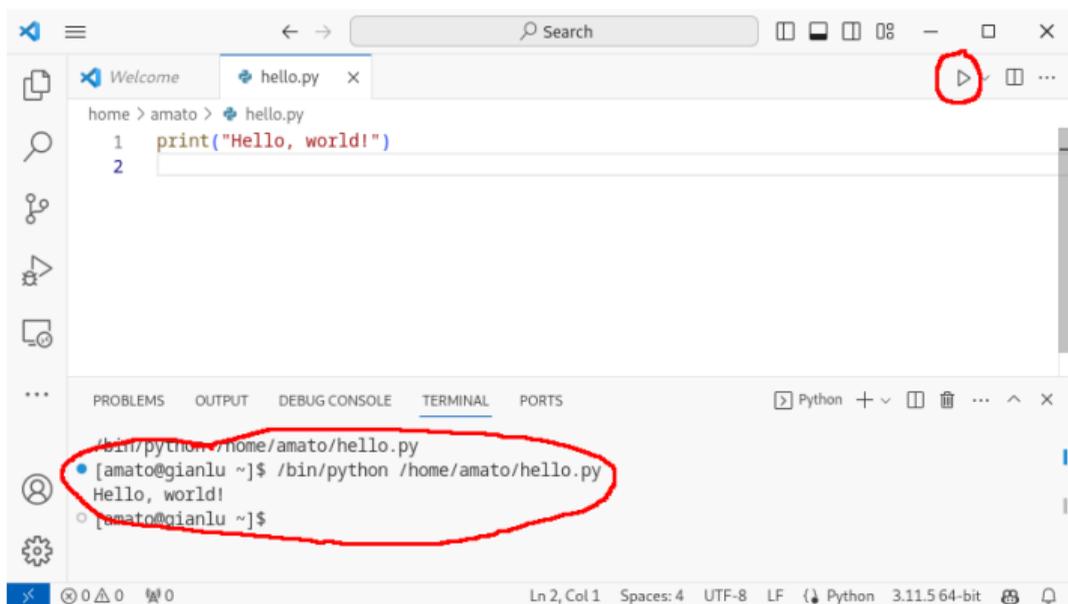
Il terminale e la shell

Interazione con Python tramite terminale

L'ambiente di sviluppo Visual Studio Code

Interazione con Python tramite Visual Studio Code

- Scrivere un file con estensione `.py` in Visual Studio Code.
- Premere il pulsante di esecuzione (in alto a destra).
- L'output viene mostrato nella parte inferiore della finestra, nel tab *Terminal*.



The screenshot shows the Visual Studio Code interface. At the top, a file named `hello.py` is open, containing the following code:

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

In the top right corner of the editor window, a red circle highlights the Run button (a play icon).

At the bottom, the Terminal panel is active, showing the command `/bin/python /home/amato/hello.py` being executed. The output of the command is `Hello, world!`. A red circle highlights the entire terminal output area.

The status bar at the bottom indicates the current file is `hello.py`, the encoding is `UTF-8`, and the Python interpreter is `Python 3.11.5 64-bit`.

## Interazione

Gianluca Amato

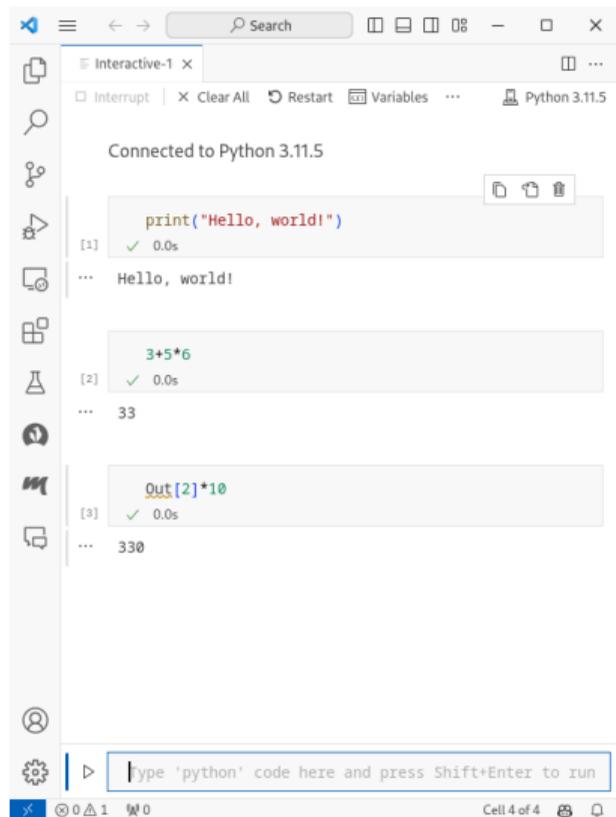
Il terminale e la shell

Interazione con Python tramite terminale

L'ambiente di sviluppo Visual Studio Code

Interazione con Python tramite Visual Studio Code

- Dare il comando *Jupyter: Create Interactive Window*
- Si apre una nuova finestra con due parti:
  - in basso, il prompt di python;
  - in alto, l'output dei comandi dati in precedenza.
- Digitare un comando e, quando si è terminato, premere la combinazione `Shift` + `Enter`.
- Il risultato appare nella parte superiore.



## Interazione

Gianluca Amato

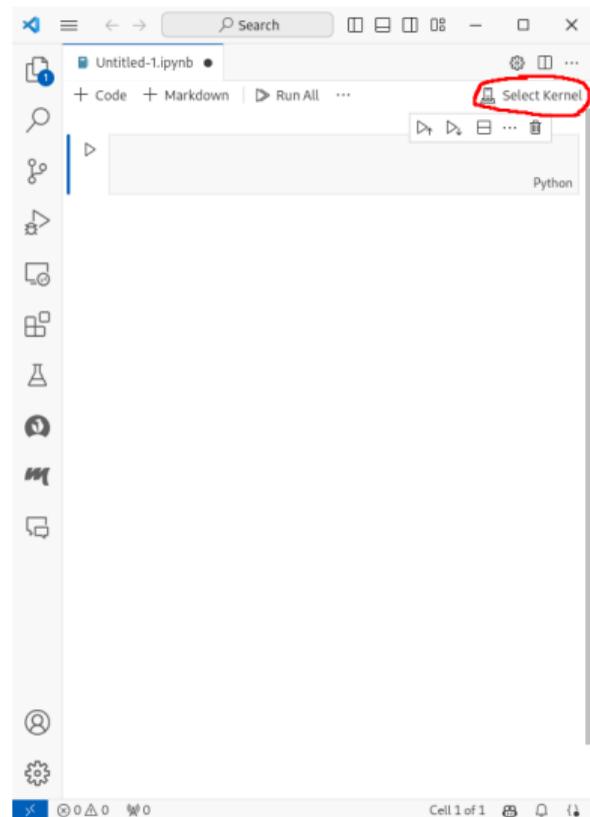
Il terminale e la shell

Interazione con Python tramite terminale

L'ambiente di sviluppo Visual Studio Code

Interazione con Python tramite Visual Studio Code

- Dare il comando *Creare: Create Jupyter Notebook*
- Si apre una finestra simile alla precedente ma:
  - non c'è lo spazio riservato al prompt;
  - si scrive direttamente nella parte che prima era l'output.
- Se necessario, cliccare *Select Kernel* in alto a destra e scegliere:
  - prima, *Python Environments. . .* ;
  - quindi, una qualunque versione di Python 3.



## Interazione

Gianluca Amato

Il terminale e la shell

Interazione con Python tramite terminale

L'ambiente di sviluppo Visual Studio Code

Interazione con Python tramite Visual Studio Code

- Digitare un comando e, quando si è terminato, premere la combinazione `Shift` + `Enter`.
- È possibile modificare e rieseguire i comandi dati in precedenza.
- In alternativa, è possibile caricare un notebook già pronto (file con estensione `.pynb`)

