



UNIVERSITÀ  
DEGLI STUDI  
DI PALERMO

CORSO DI LAUREA IN **INGEGNERIA ROBOTICA**

CORSO DI **PROGRAMMAZIONE DI ROBOT C.I.**

PROF. ROSARIO SORBELLO LEZIONE DEL 2 OTTOBRE 2024

*Python*

# Introduzione e installazione

# Introduzione a Python

- Python è un linguaggio di programmazione **interpretato** di alto livello, moderno, che sfrutta una sintassi semplice e, per certi versi, intuitiva
- Si contraddistingue per una sintassi molto compatta
- Fu ideato da Guido van Rossum negli anni '90



*Guida van Rossum, 1956*

# Introduzione a Python

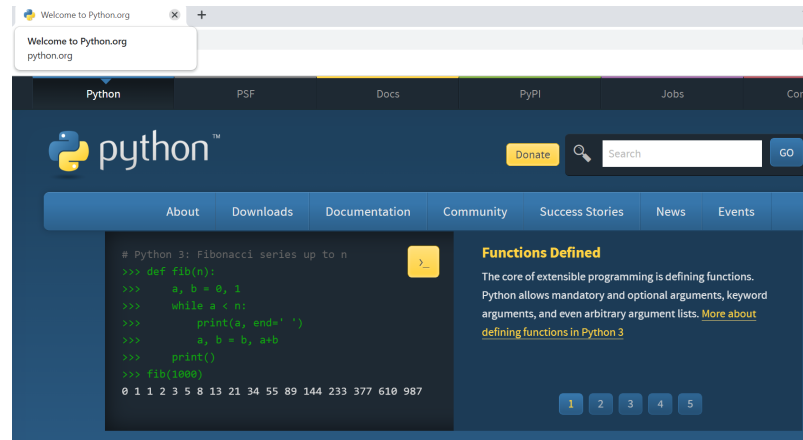
- Python si contraddistingue per una **sintassi molto compatta**
- È un linguaggio molto versatile: può essere utilizzato sia per semplici script, sia per programmi complessi, eventualmente sfruttando anche il paradigma della **programmazione a oggetti**
- Per i nostri scopi, lo utilizzeremo per implementare, in pratica, semplici programmi di esempio, e mettere in pratica la teoria sugli algoritmi introdotta in precedenza

# Introduzione a Python

- Ad oggi, l'ultima versione stabile di Python è la **3.10**, anche se la versione 2.7 sia molto usata.
- In queste slide si farà riferimento a Microsoft Windows, principalmente alla versione 10. Ciononostante, è comunque possibile installare Python e gli strumenti che utilizzeremo anche su Mac OS o Linux

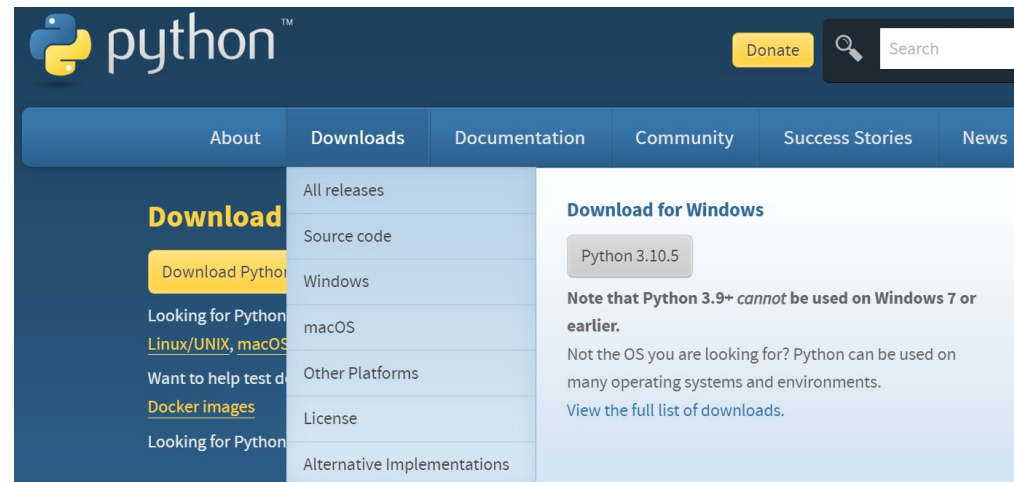
# Introduzione a Python

- Il fatto che Python sia un linguaggio interpretato ci consente di eseguire i programmi realizzati con questo linguaggio su qualsiasi sistema operativo, purché sia installato l'**interprete Python**
- Sia l'interprete che i molti strumenti di supporto a Python sono gratuiti e disponibili per il download tramite il sito ufficiale [python.org](https://python.org)



# Installare Python

- Dal menu della pagina principale selezionare Downloads
- Sarà messa in evidenza l'ultima versione stabile per il vostro sistema operativo
- Altre versioni potranno essere scaricate cliccando sul sistema operativo desiderato



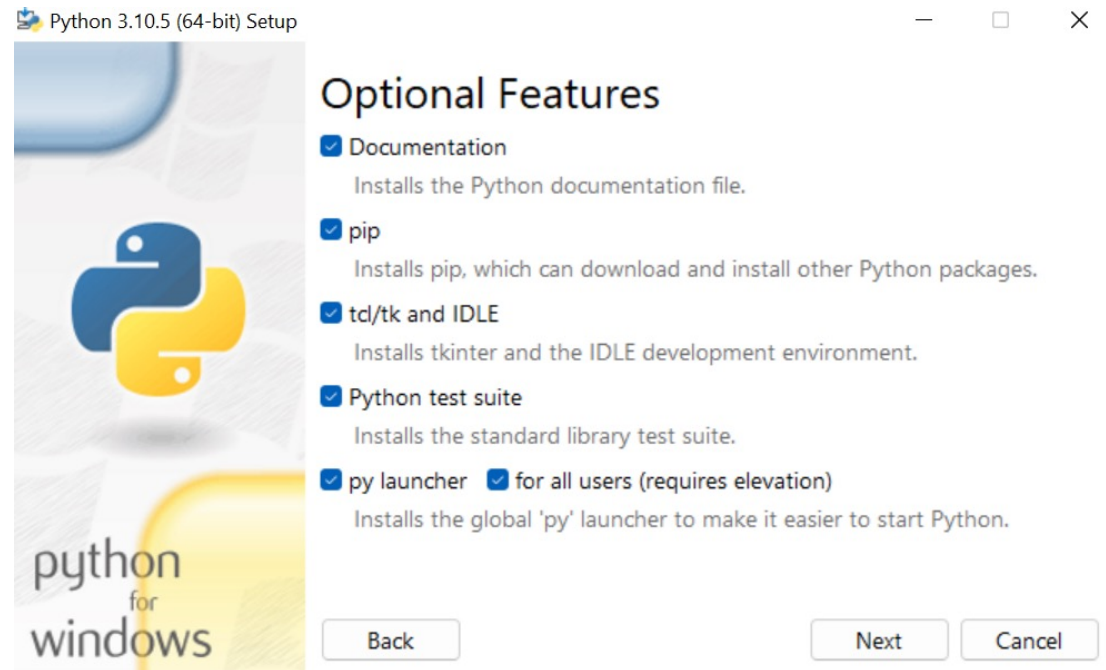
# Installare Python

- Una volta scaricato il file eseguibile, avviamolo e procediamo con l'installazione
- **IMPORTANTE!** Assicurarsi di selezionare le voci:
  - *Install launcher for all users*
  - *Add Python 3.6 to PATH*
- Clicchiamo ora su *Install Now*



# Installare Python

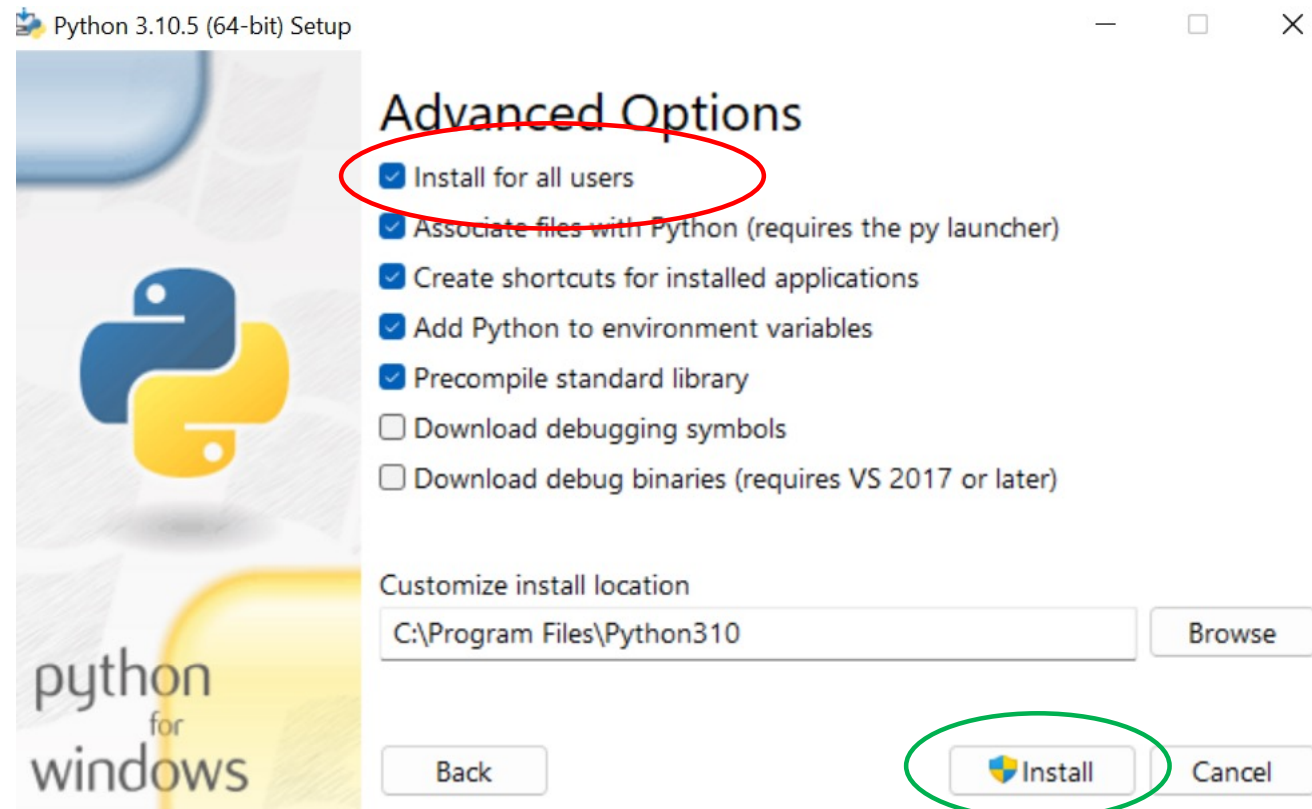
- Se non sono già selezionati per default, selezionare tutte le opzioni per avere la suite completa.





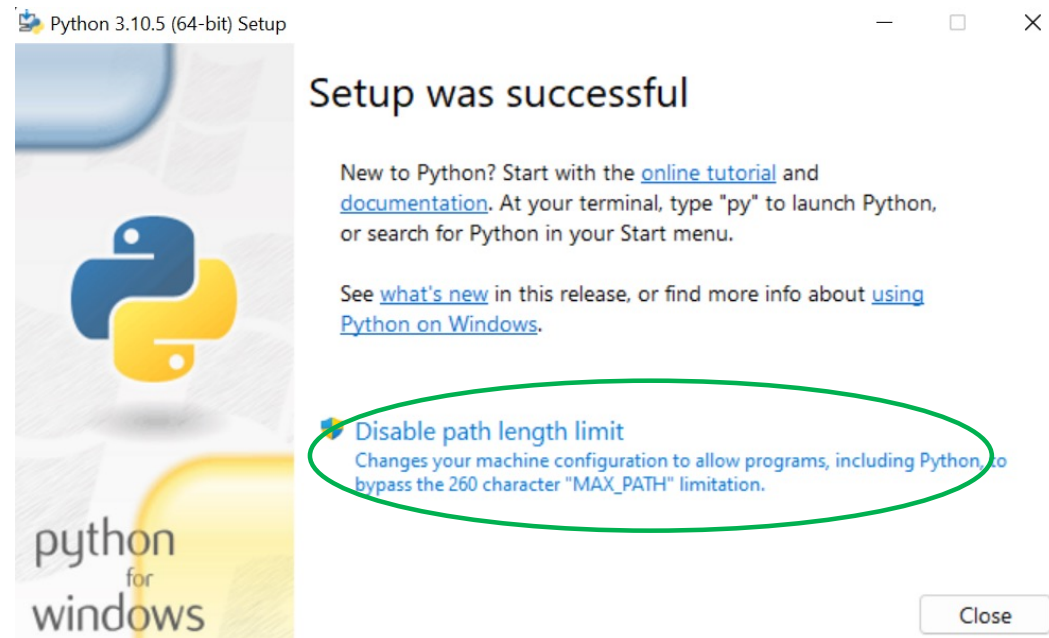
# Installare Python

- Selezionare la prima opzione: per tutti gli utenti
- A questo punto si può procedere con l'installazione
- Al successivo messaggio se si vuole installare Python, rispondete sì.



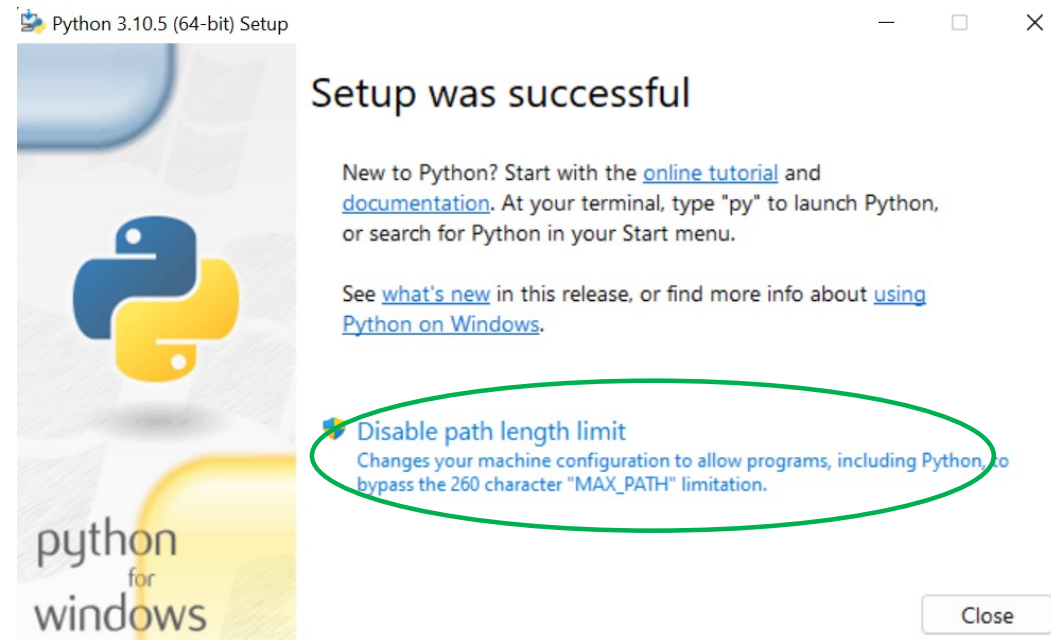
# Installare Python

- Se l'installazione è andata a buon fine, apparirà la seguente finestra di messaggio
- Si notano un link ad un tutorial online:  
<https://docs.python.org/3.10/tutorial/index.html>
- La documentazione:  
<https://docs.python.org/3.10/index.html>
- ...ed altri link utili.





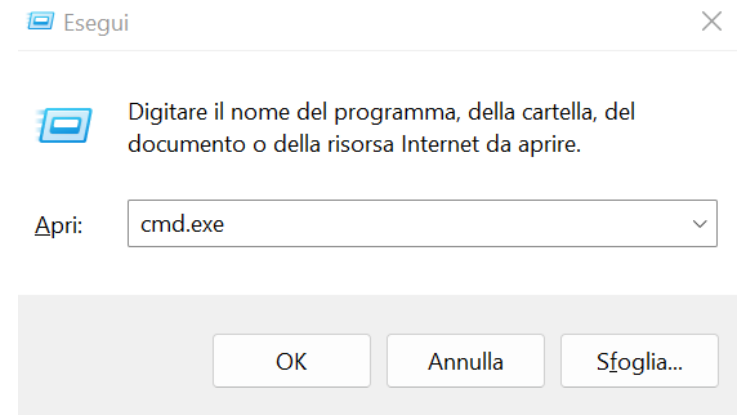
# Installare Python

- Selezionare la disattivazione della lunghezza massima dei path

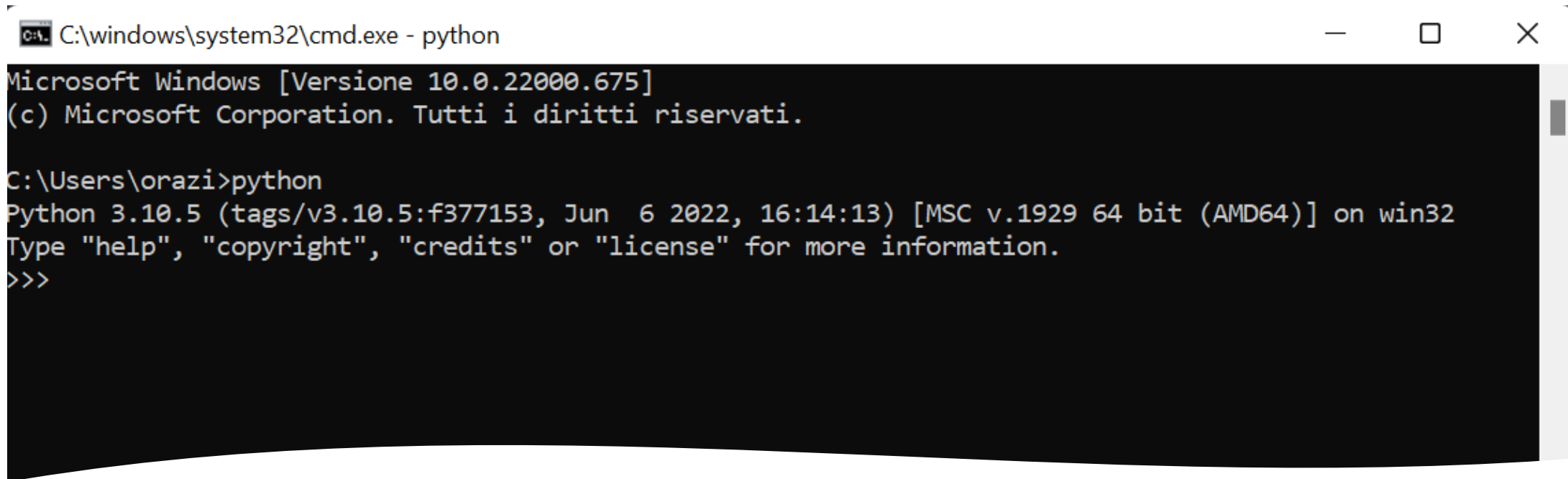


# Installare Python

- Al termine dell'installazione, se abbiamo correttamente seguito tutte le indicazioni, Python sarà stato correttamente installato
- Per verificarlo, apriamo il prompt dei comandi di Windows:
  - Per farlo, premiamo i tasti  + 
  - Nella finestra che si apre, digitiamo la stringa *cmd.exe*, e premiamo *Invio*
- Digitiamo ora *python* nel prompt dei comandi, e premiamo *Invio*



# L'interprete Python

A screenshot of a Windows command prompt window. The title bar at the top reads 'C:\windows\system32\cmd.exe - python'. The window has standard Windows window controls (minimize, maximize, close) on the right. The command prompt shows the following text: 'Microsoft Windows [Versione 10.0.22000.675]' followed by '(c) Microsoft Corporation. Tutti i diritti riservati.' on the next line. The user has entered the command 'python' at the prompt 'C:\Users\orazi>'. The output shows 'Python 3.10.5 (tags/v3.10.5:f377153, Jun 6 2022, 16:14:13) [MSC v.1929 64 bit (AMD64)] on win32' followed by 'Type "help", "copyright", "credits" or "license" for more information.' and the prompt '>>>' on the next line.

```
C:\windows\system32\cmd.exe - python

Microsoft Windows [Versione 10.0.22000.675]
(c) Microsoft Corporation. Tutti i diritti riservati.

C:\Users\orazi>python
Python 3.10.5 (tags/v3.10.5:f377153, Jun 6 2022, 16:14:13) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

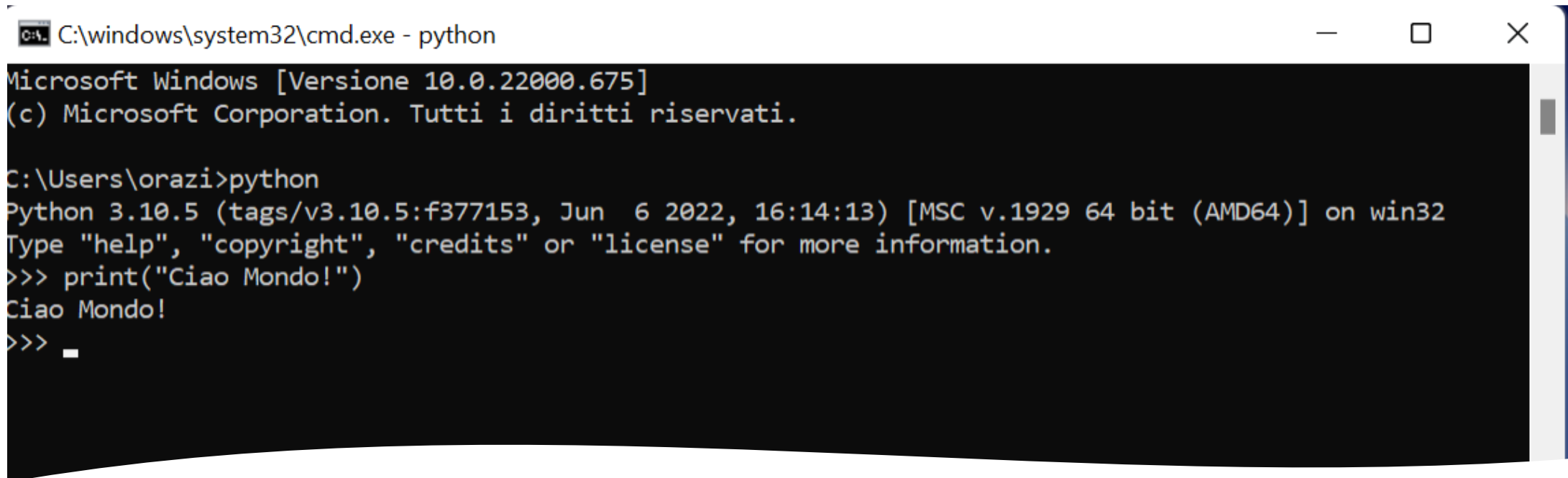
- Se l'installazione è andata a buon fine, si aprirà l'interprete Python

# Hello World!



[https://it.wikipedia.org/wiki/Ada\\_Lovelace](https://it.wikipedia.org/wiki/Ada_Lovelace)

# L'interprete Python

A screenshot of a Windows command prompt window. The title bar at the top reads 'C:\windows\system32\cmd.exe - python'. The window content shows the following text: 'Microsoft Windows [Versione 10.0.22000.675] (c) Microsoft Corporation. Tutti i diritti riservati.' followed by a prompt 'C:\Users\orazi>python'. The output shows 'Python 3.10.5 (tags/v3.10.5:f377153, Jun 6 2022, 16:14:13) [MSC v.1929 64 bit (AMD64)] on win32' and 'Type "help", "copyright", "credits" or "license" for more information.' Then, the user enters '>>> print("Ciao Mondo!")' and the output 'Ciao Mondo!' is displayed. The prompt '>>> \_' is shown at the bottom.

```
C:\windows\system32\cmd.exe - python

Microsoft Windows [Versione 10.0.22000.675]
(c) Microsoft Corporation. Tutti i diritti riservati.

C:\Users\orazi>python
Python 3.10.5 (tags/v3.10.5:f377153, Jun 6 2022, 16:14:13) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Ciao Mondo!")
Ciao Mondo!
>>> _
```

- Tramite l'interprete, possiamo già eseguire le prime istruzioni, come la funzione di visualizzazione su schermo **print**

# L'interprete Python

- Supponiamo ad esempio di scrivere una serie di istruzioni Python e le scriviamo con un qualsiasi editor (anche il Blocco Note di Windows va bene!)

hello.py

→ L'estensione *.py* è tipica degli script Python

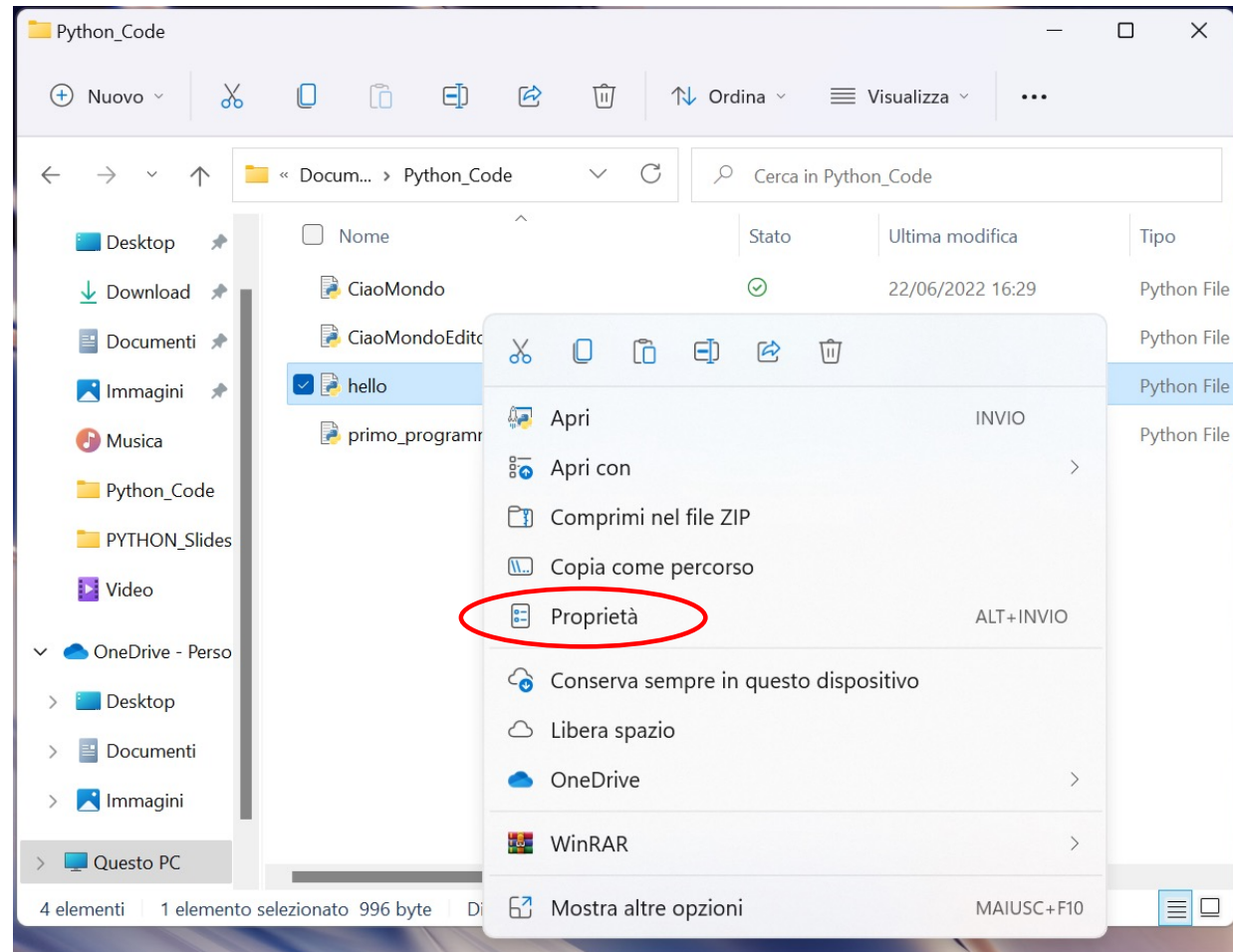
```
def main(args):  
    print("Ciao Mondo!")  
    return 0  
  
if __name__ == '__main__':  
    import sys  
    sys.exit(main(sys.argv))
```

- Tutte le istruzioni contenute nel file *hello.py* verranno eseguite, senza bisogno di doverle digitare una per una
- Il file è salvato nella cartella:  
*Documenti -> Python\_Code*
- Nelle prossime slides vediamo come fare ad eseguire il codice tramite interprete python dalla Shell di Windows.



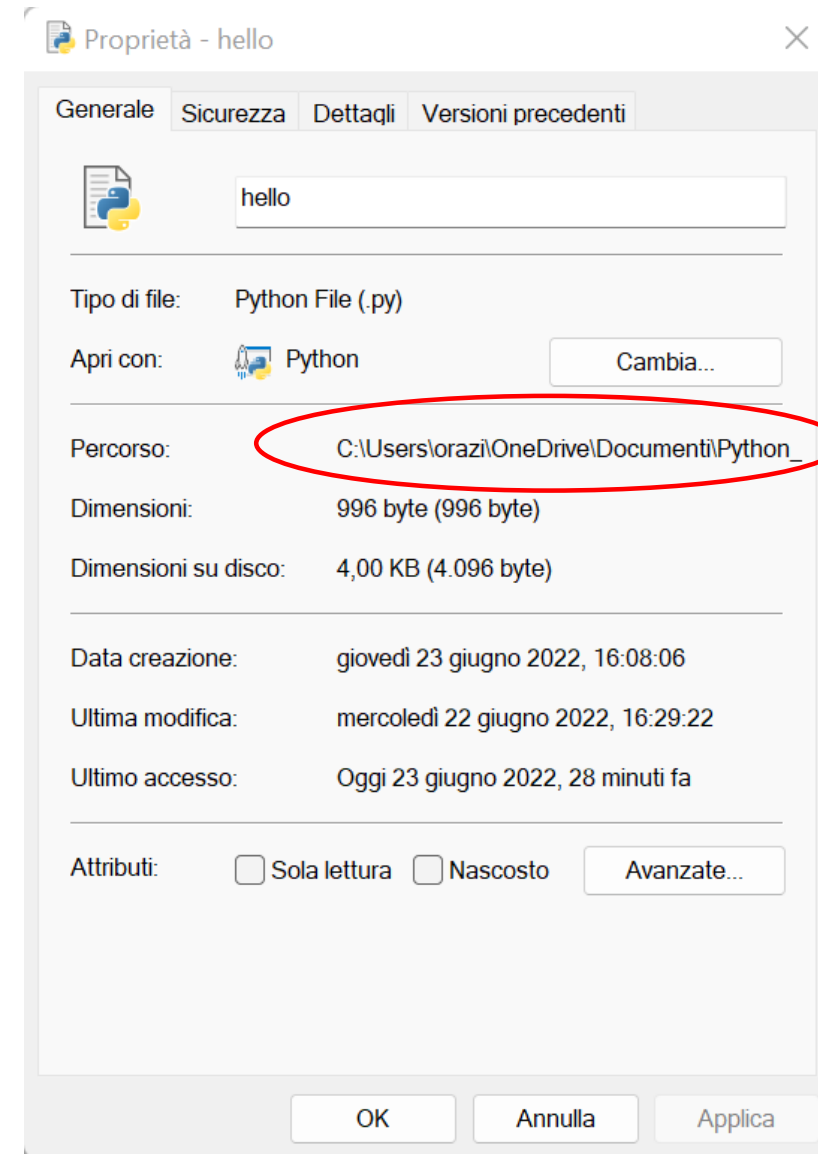
# L'interprete Python

- Per ricavare il path al file *hello.py* è sufficiente cliccare su Proprietà del menu del tasto destro puntando il file con il puntatore

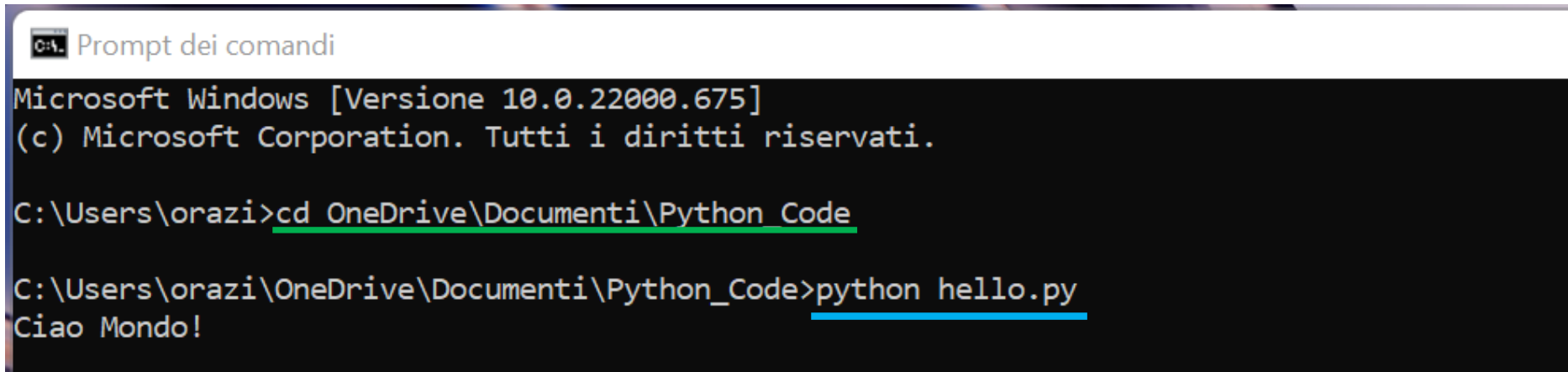


# L'interprete Python

- Il path al file hello.py è evidenziato in rosso.



# L'interprete Python



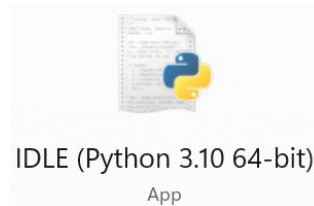
```
C:\> Prompt dei comandi
Microsoft Windows [Versione 10.0.22000.675]
(c) Microsoft Corporation. Tutti i diritti riservati.

C:\Users\orazi>cd OneDrive\Documenti\Python Code

C:\Users\orazi\OneDrive\Documenti\Python_Code>python hello.py
Ciao Mondo!
```

- E' necessario prima posizionarsi sulla cartella Python\_Code, dove si trovano tutti i codici .py che scriveremo, tramite il comando *cd*
- Quindi digitare *python hello.py*
- Nella riga successiva apparirà "Ciao Mondo!"

# La *Shell* di Python




- Una *Shell* è un programma che rende disponibili, tramite opportuni comandi ed istruzioni, i servizi di un sistema operativo o la scrittura del codice per un interprete di linguaggio ad un utente umano
- In generale, le shell utilizzano un'interfaccia a riga di comando (Command-Line Interface, CLI) o un'interfaccia grafica (Graphical User Interface, GUI) a seconda del ruolo del computer e dell'operazione particolare
- Si chiama *Shell* perché è lo strato più esterno attorno al sistema operativo
- Il prompt del sistema operativo Windows (*cmd.exe*) è una *Shell*

# Hello World!

- Per iniziare a familiarizzare con un linguaggio di programmazione, in genere si parte da un esempio molto semplice di codice
- L'esempio più semplice e più utilizzato è comunemente identificato come *Hello World!* (*Ciao Mondo!*), e identifica un semplice programma che ha, come unico scopo, la stampa a video di una stringa (in genere "Hello World!", appunto)
- In Python, tutto ciò si realizza con un'unica riga di codice:

```
print("hello world")
```

# La *Shell* di Python

- La *Shell* di Python permette di scrivere direttamente il codice Python senza accedere al prompt dei comandi e senza digitare *python* per attivare l'interprete del linguaggio
- Per avviare la *Shell* di Python, premere il tasto , o l'icona di Windows, e cercare "idle"
- Cliccando sull'icona comparirà la Shell di Python mostrata qui a destra

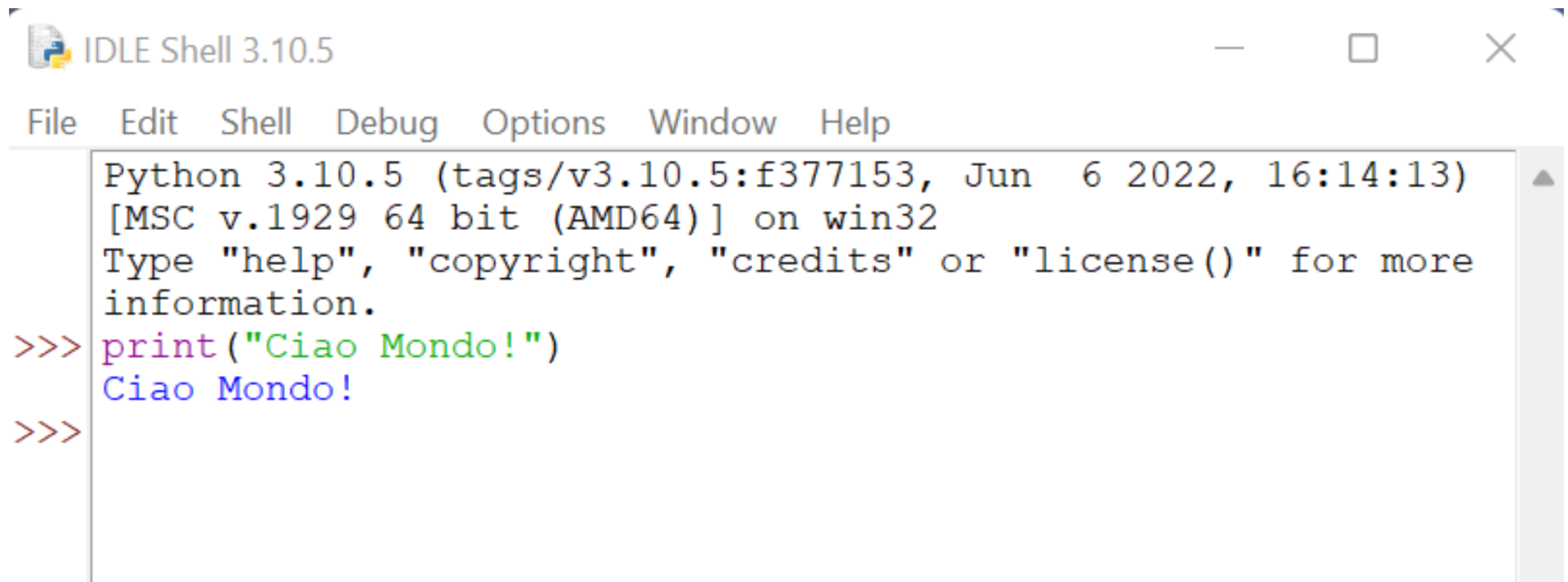


```
Python 3.10.5 (tags/v3.10.5:f377153, Jun 6 2022, 16:14:13)
[MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more
information.
>>>
```

Ln: 3 Col: 0

22

# Ciao Mondo! con la Shell di Python



```
Python 3.10.5 (tags/v3.10.5:f377153, Jun 6 2022, 16:14:13)
[MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more
information.
>>> print("Ciao Mondo!")
Ciao Mondo!
>>>
```

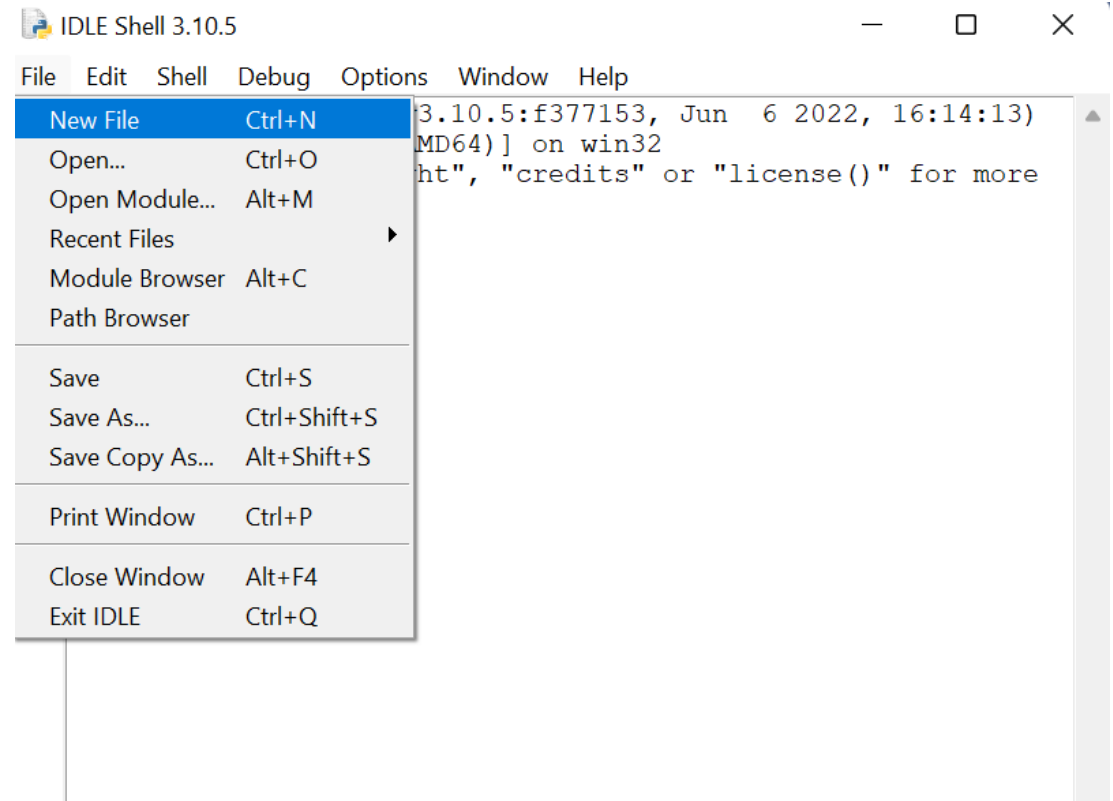
# L'interprete Python

- una Shell permette di eseguire solamente una singola riga di istruzione alla volta.
- La Shell è comoda se vogliamo iniziare a prendere confidenza con Python, ad esempio provando ad eseguire qualche semplice operazione matematica
- Per la scrittura di programmi in un qualsiasi linguaggio è indispensabile un editor particolare chiamato IDE (Integrated Development Environment) che nel Python è chiamato IDLE (Integrated Development and Learning Environment)
- L'editor per un linguaggio di programmazione è indispensabile per la scrittura di programmi.

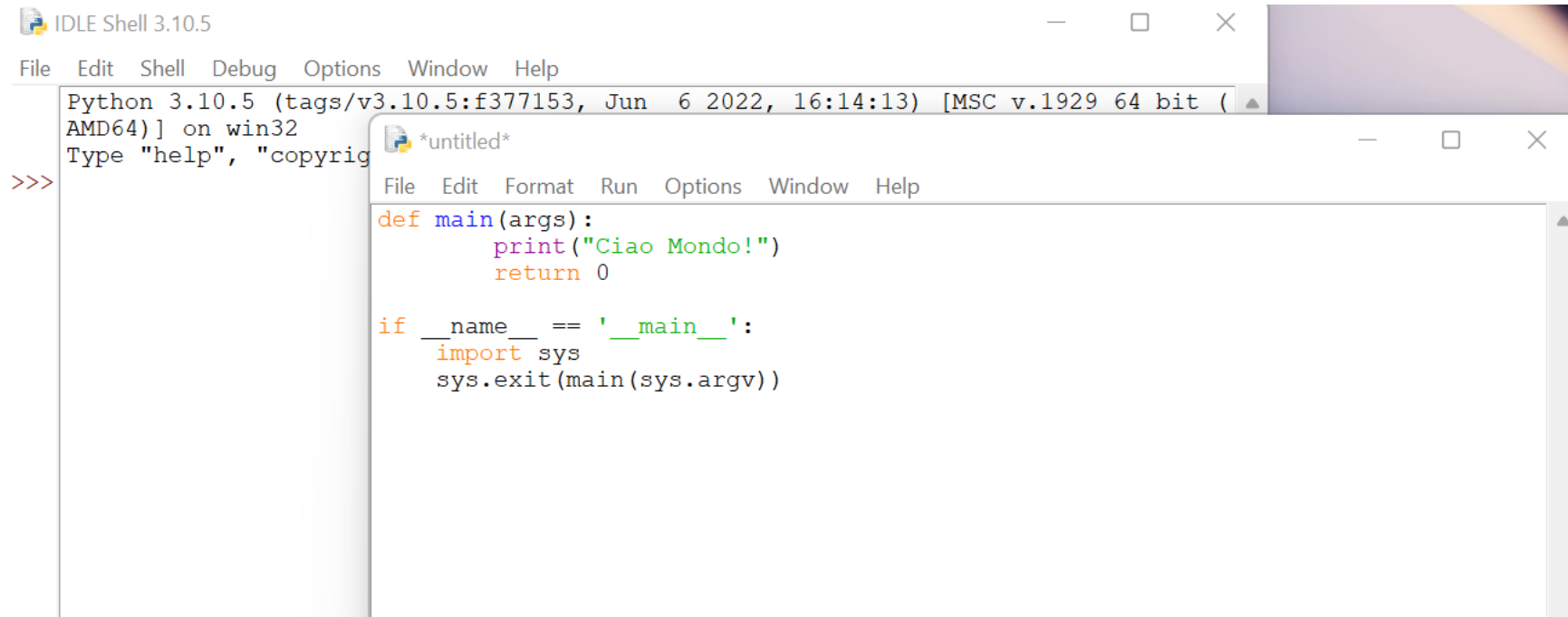


# Editor Python integrato nella *Shell*

- Se non volete installare un editor dedicato alla scrittura del codice, potete sempre usare quello integrato nella Shell
- Per avviarlo, basta andare su *File -> New File*

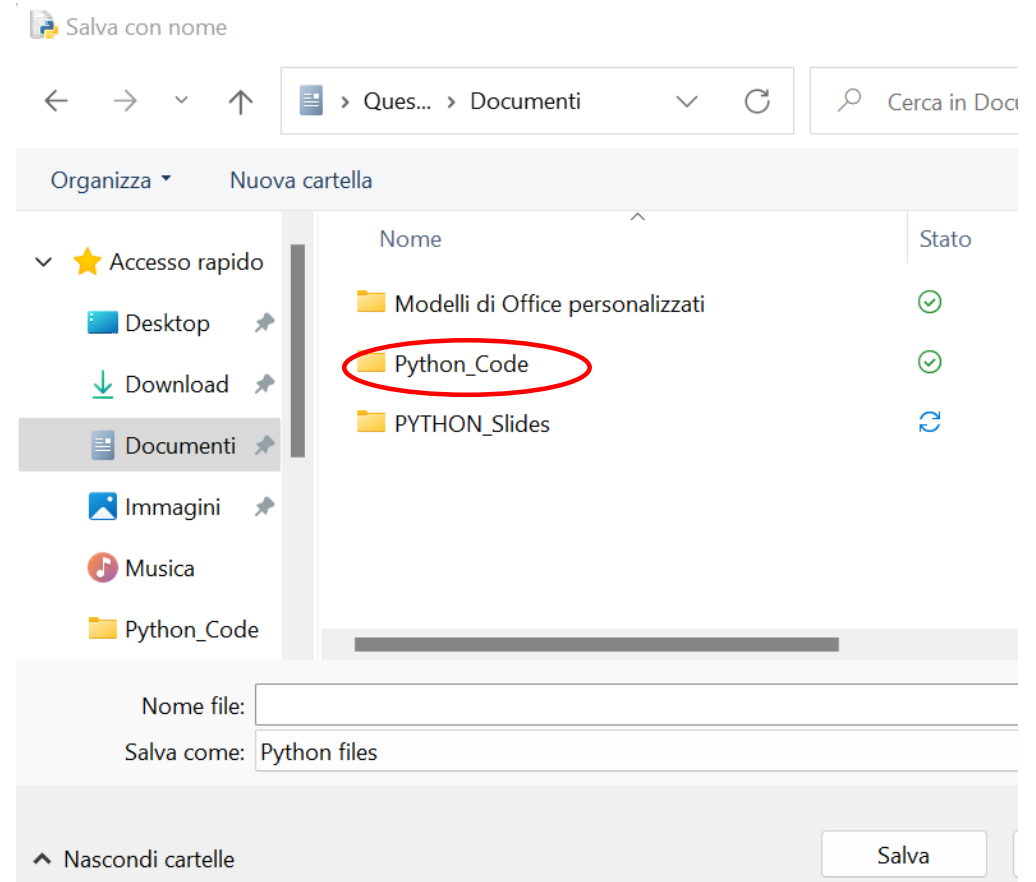


# Editor Python integrato nella *Shell*



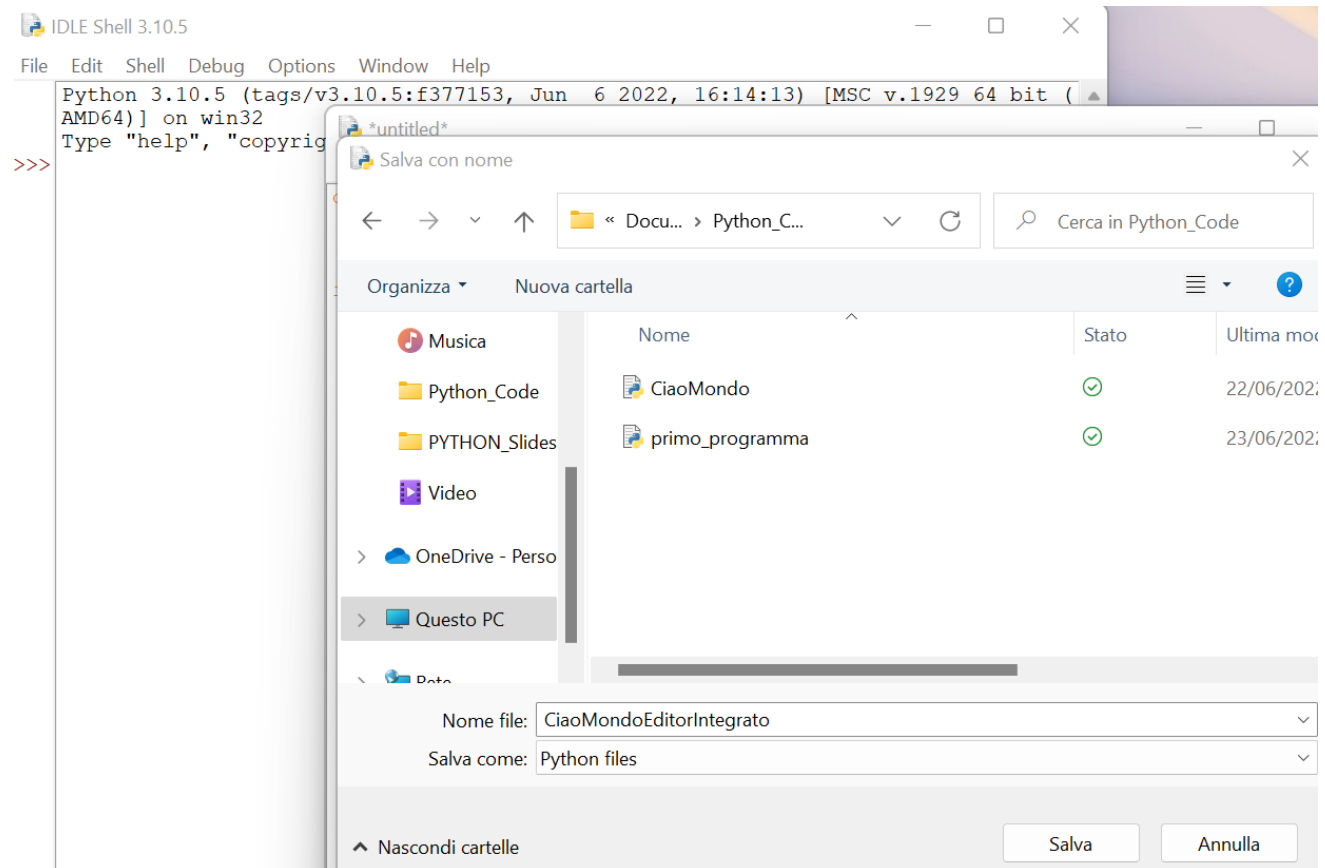
# Editor Python integrato nella *Shell*

- E' fortemente consigliata la creazione di una cartella apposita in *Documenti* in cui conservare il codice che scrivete
- Potete chiamare la cartella con il nome che più vi piace, ma evitate di usare spazi. Al posto dello spazio usate il simbolo \_ [shift -]
- In questo caso l'ho chiamata Python\_Code

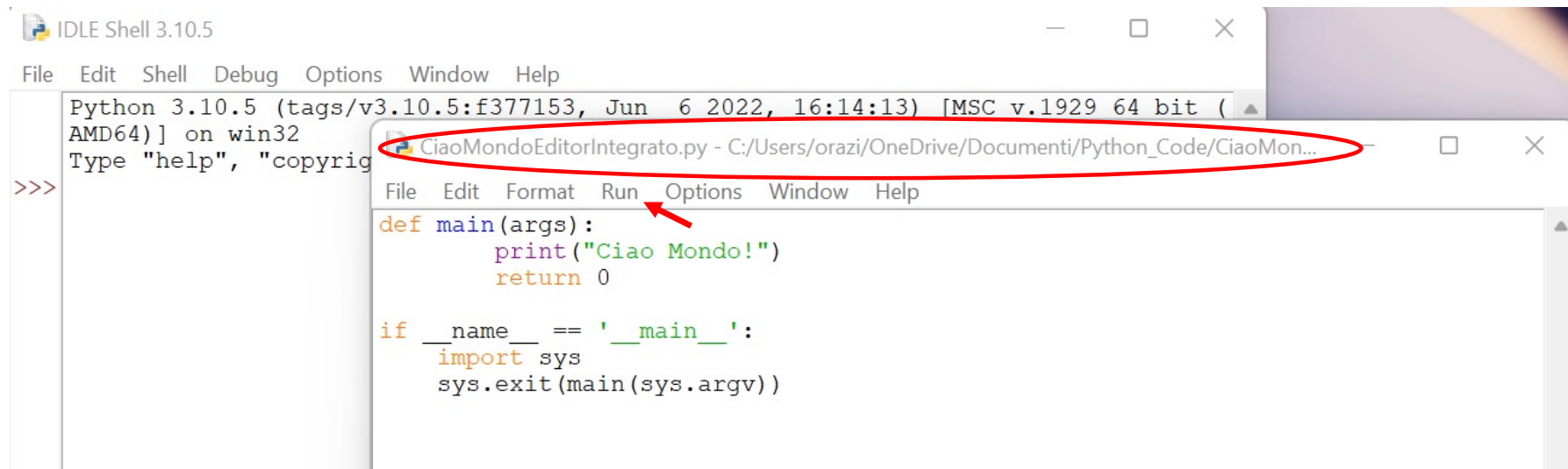


# Editor Python integrato nella *Shell*

- Salvate il file .py con il nome che più vi piace.
- Per esempio:
  - CiaoMondoEditorIntegrato

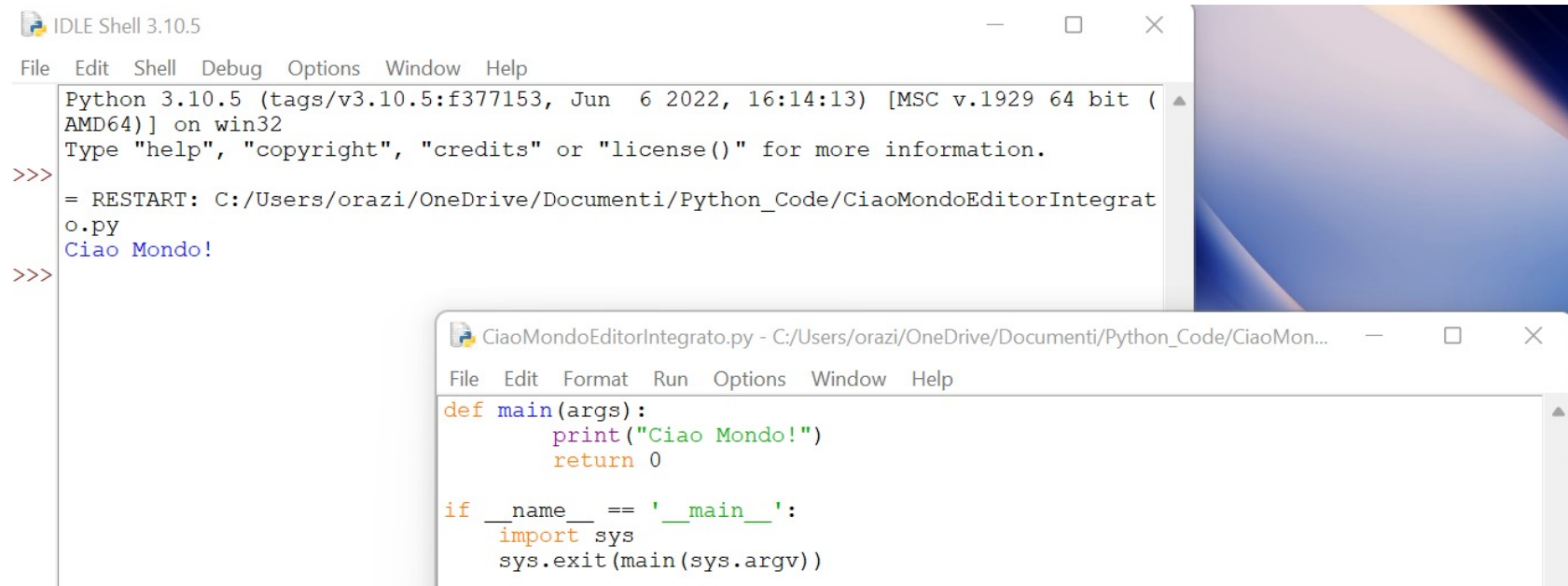


# Editor Python integrato nella *Shell*



- Una volta salvato, il nome del file comparirà sulla barra dell'editor in alto con accanto il path per raggiungerlo.
- Per eseguire il programma, cliccate con il mouse *Run->Run Module*

# Editor Python integrato nella *Shell*



The image shows two overlapping windows from the IDLE Python environment. The background window is the 'IDLE Shell 3.10.5' with a menu bar (File, Edit, Shell, Debug, Options, Window, Help). It displays the Python 3.10.5 startup message and a prompt 'Ciao Mondo!' that has been executed. The foreground window is a script editor titled 'CiaoMondoEditorIntegrato.py - C:/Users/orazi/OneDrive/Documenti/Python\_Code/CiaoMon...'. It contains a Python script with a `main` function that prints 'Ciao Mondo!' and returns 0, and a standard `__main__` guard.

```
Python 3.10.5 (tags/v3.10.5:f377153, Jun 6 2022, 16:14:13) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/orazi/OneDrive/Documenti/Python_Code/CiaoMondoEditorIntegrato.py
Ciao Mondo!
>>>
```

```
def main(args):
    print("Ciao Mondo!")
    return 0

if __name__ == '__main__':
    import sys
    sys.exit(main(sys.argv))
```

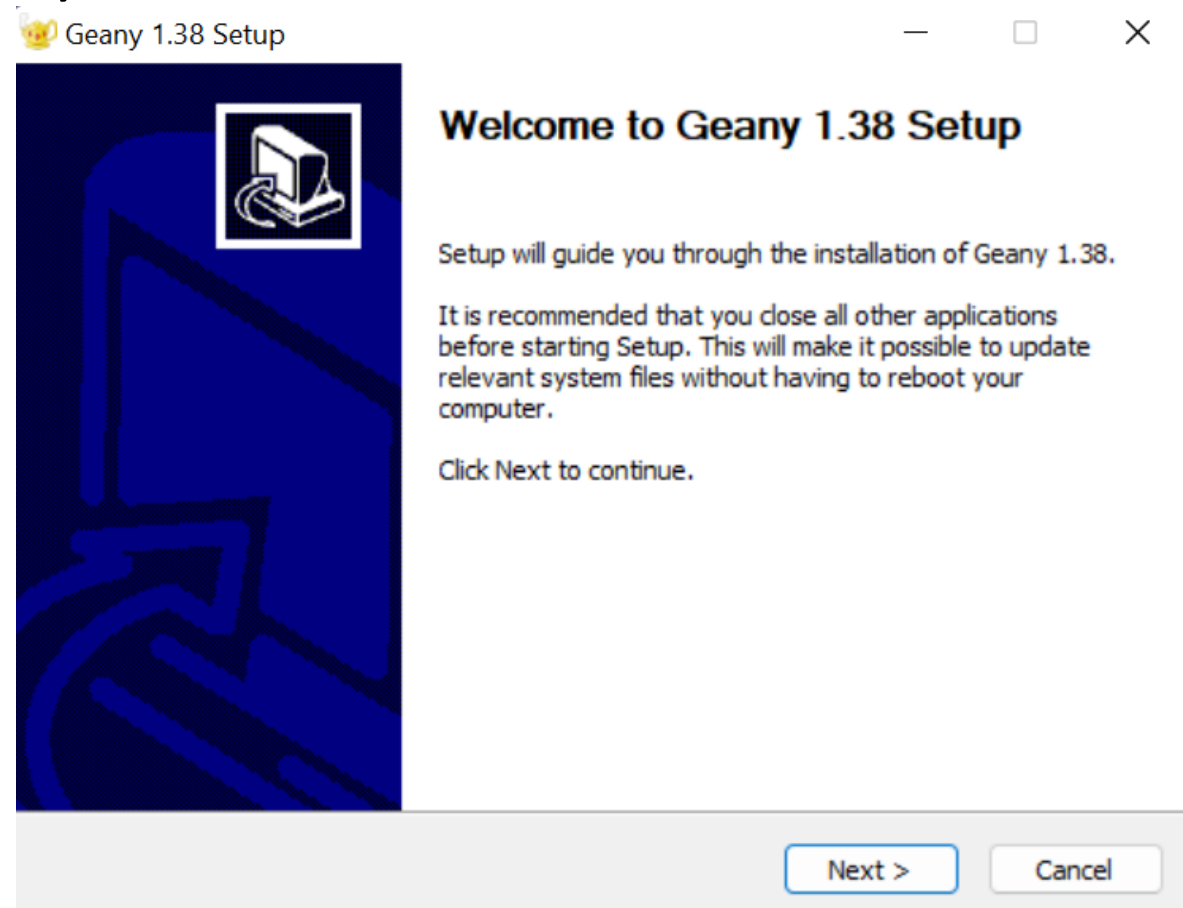
- Nella Shell dalla quale abbiamo aperto l'editor comparirà l'esecuzione del programma.

# L'editor di testo Geany

- Per scrivere uno script in Python, possiamo utilizzare l'editor di testo integrato appena vista, ma anche qualsiasi altro editor (incluso il Blocco Note di Windows)
- Tuttavia, alcuni editor ci permettono di sfruttare funzionalità che semplificano sia la scrittura degli script, sia l'esecuzione di essi
- Uno di questi editor è **Geany**: scaricabile al link <https://www.geany.org/download/releases/>
- Per installare Geany, dopo aver scaricato *geany-1.38\_setup.exe*, eseguiamo il file

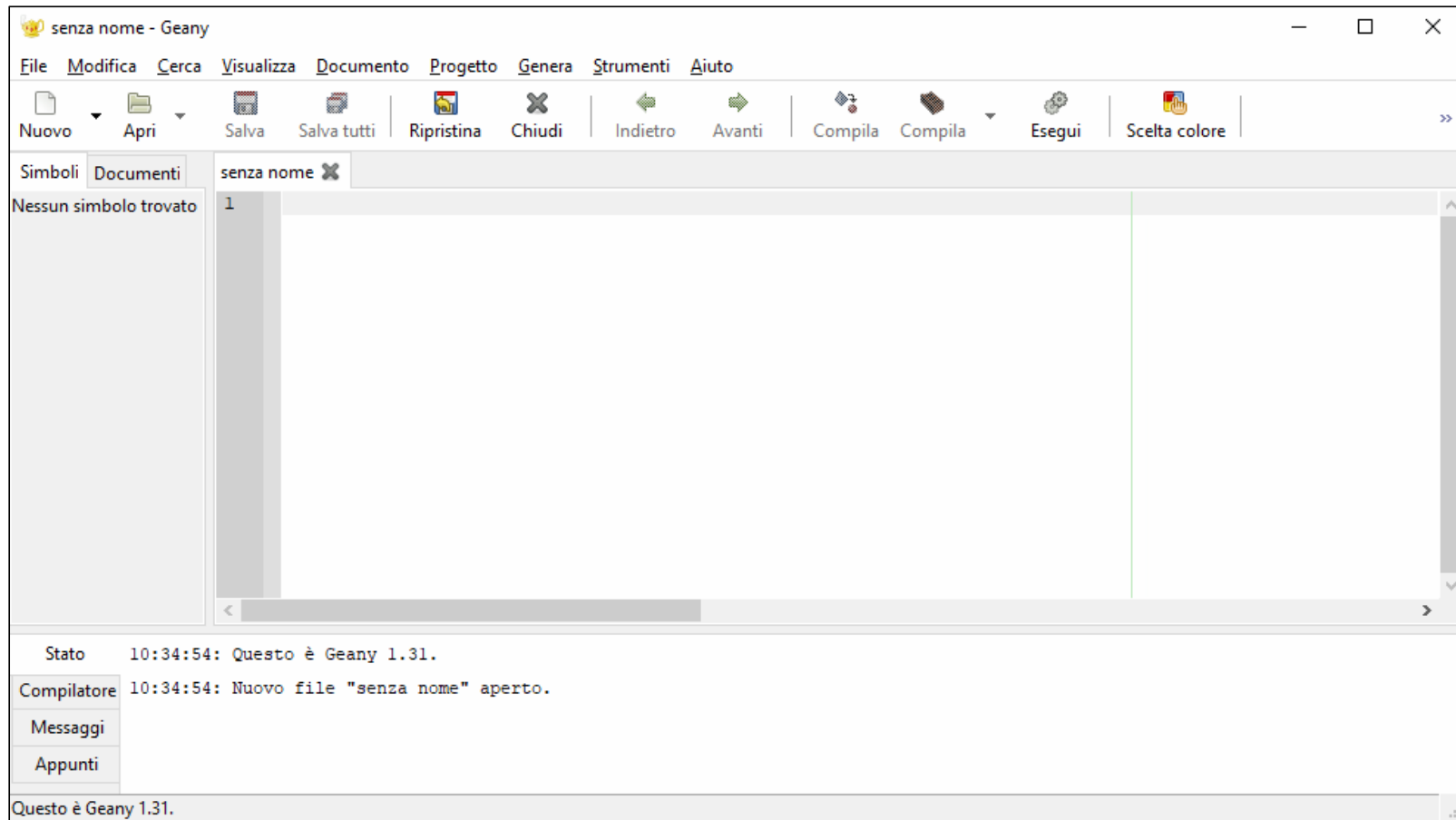
# L'editor di testo Geany

- Effettuiamo l'installazione, utilizzando tutte le impostazioni di default (Full)
- Una volta completata l'installazione, sul Desktop verrà visualizzata l'icona di Geany:



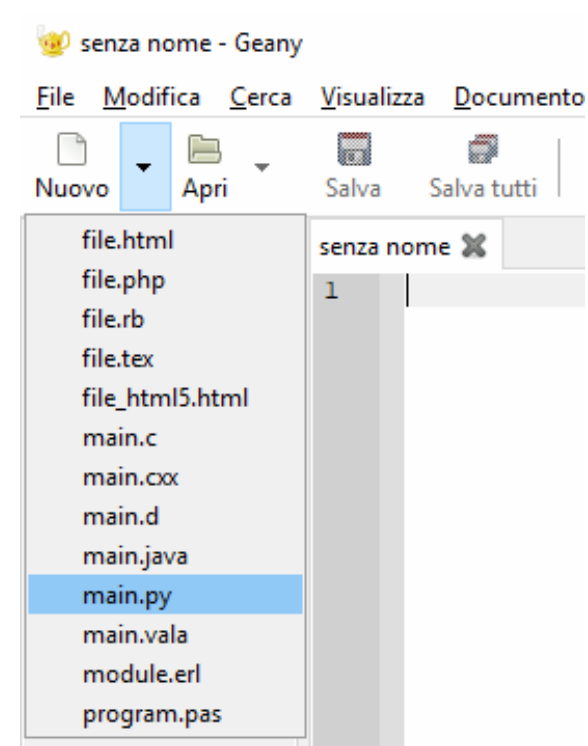


# L'editor di testo Geany



# L'editor di testo Geany

- Possiamo utilizzare Geany per creare un nuovo script Python
- Clicchiamo sulla freccia di fianco alla voce *Nuovo*, e selezioniamo *main.py*



Commenti

I commenti sono stringhe di testo, in cui ogni riga è preceduta dal carattere #

L'interprete si limita a scartare tutte le righe di codice che iniziano con #

Codice

```
1#!/usr/bin/env python
2# -*- coding: utf-8 -*-
3#
4#  senza nome.py
5#
6#  Copyright 2017 Vito Gentile <vito.gentile@unipa.it>
7#
8#  This program is free software; you can redistribute it and/or modify
9#  it under the terms of the GNU General Public License as published by
10#  the Free Software Foundation; either version 2 of the License, or
11#  (at your option) any later version.
12#
13#  This program is distributed in the hope that it will be useful,
14#  but WITHOUT ANY WARRANTY; without even the implied warranty of
15#  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
16#  GNU General Public License for more details.
17#
18#  You should have received a copy of the GNU General Public License
19#  along with this program; if not, write to the Free Software
20#  Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston,
21#  MA 02110-1301, USA.
22#
23#
24#
25#
26def main(args):
27    return 0
28
29if __name__ == '__main__':
30    import sys
31    sys.exit(main(sys.argv))
32
```

Stato 12:48:14: Questo è Geany 1.31.

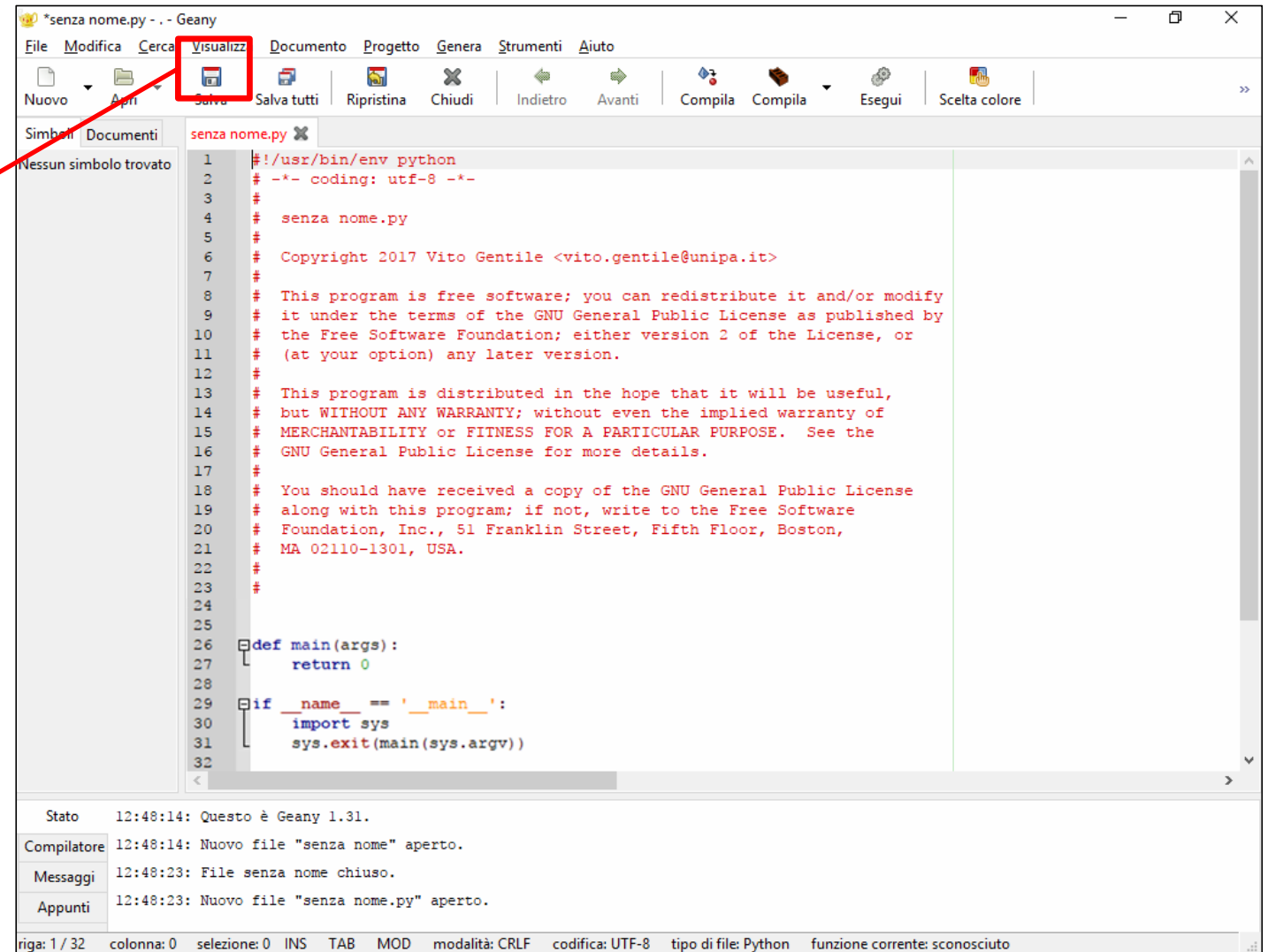
Compilatore 12:48:14: Nuovo file "senza nome" aperto.

Messaggi 12:48:23: File senza nome chiuso.

Appunti 12:48:23: Nuovo file "senza nome.py" aperto.

riga: 1 / 32 colonna: 0 selezione: 0 INS TAB MOD modalità: CRLF codifica: UTF-8 tipo di file: Python funzione corrente: sconosciuto

Salviamo il nostro script

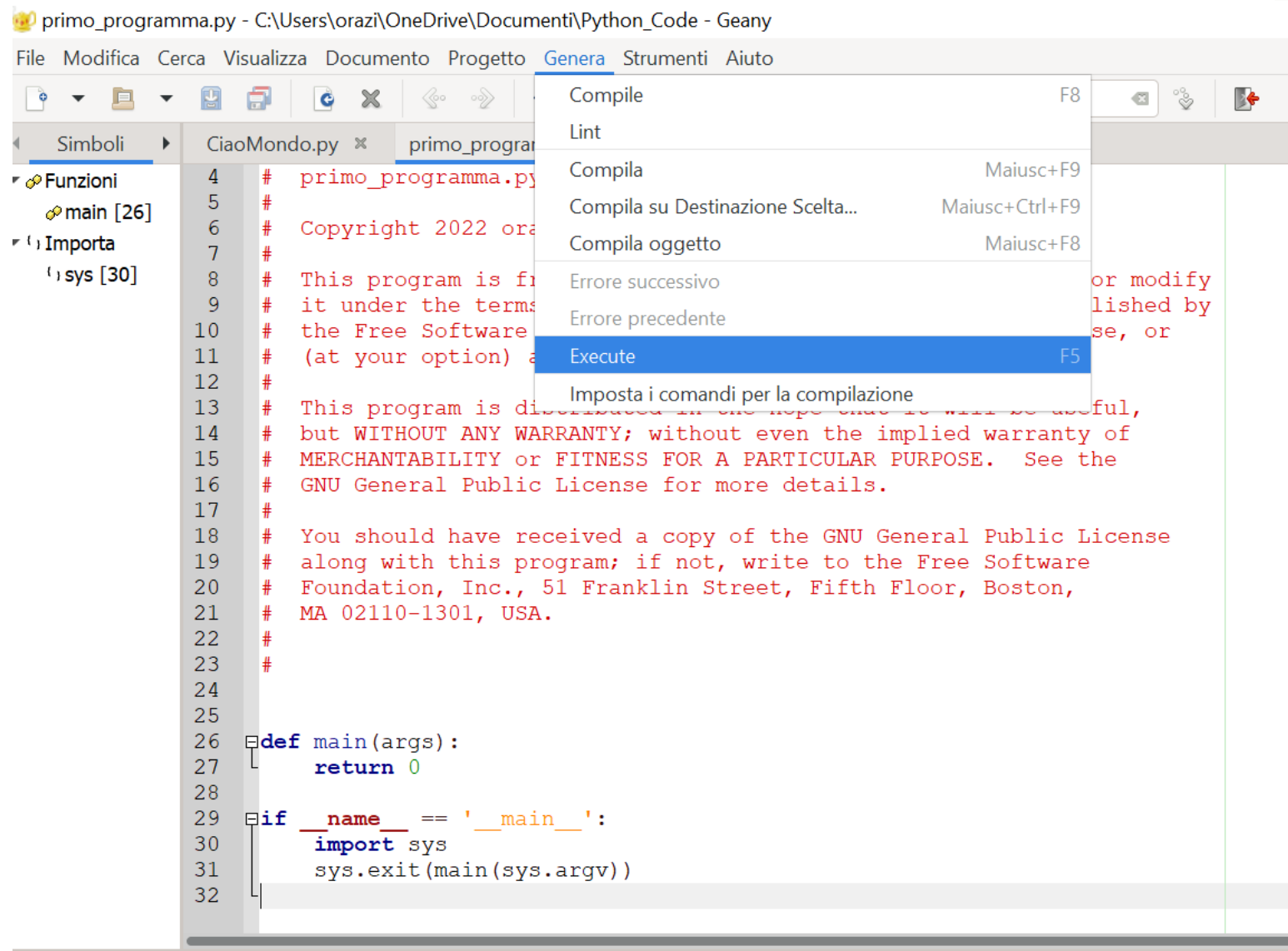


The screenshot shows the Geany IDE interface. The menu bar at the top includes 'File', 'Modifica', 'Cerca', 'Visualizz', 'Documento', 'Progetto', 'Genera', 'Strumenti', and 'Aiuto'. The 'Visualizz' menu is highlighted with a red box, and a red arrow points from the text 'Salviamo il nostro script' to the 'Salva' (Save) icon in the menu. The main editor window displays a Python script named 'senza nome.py'. The script content is as follows:

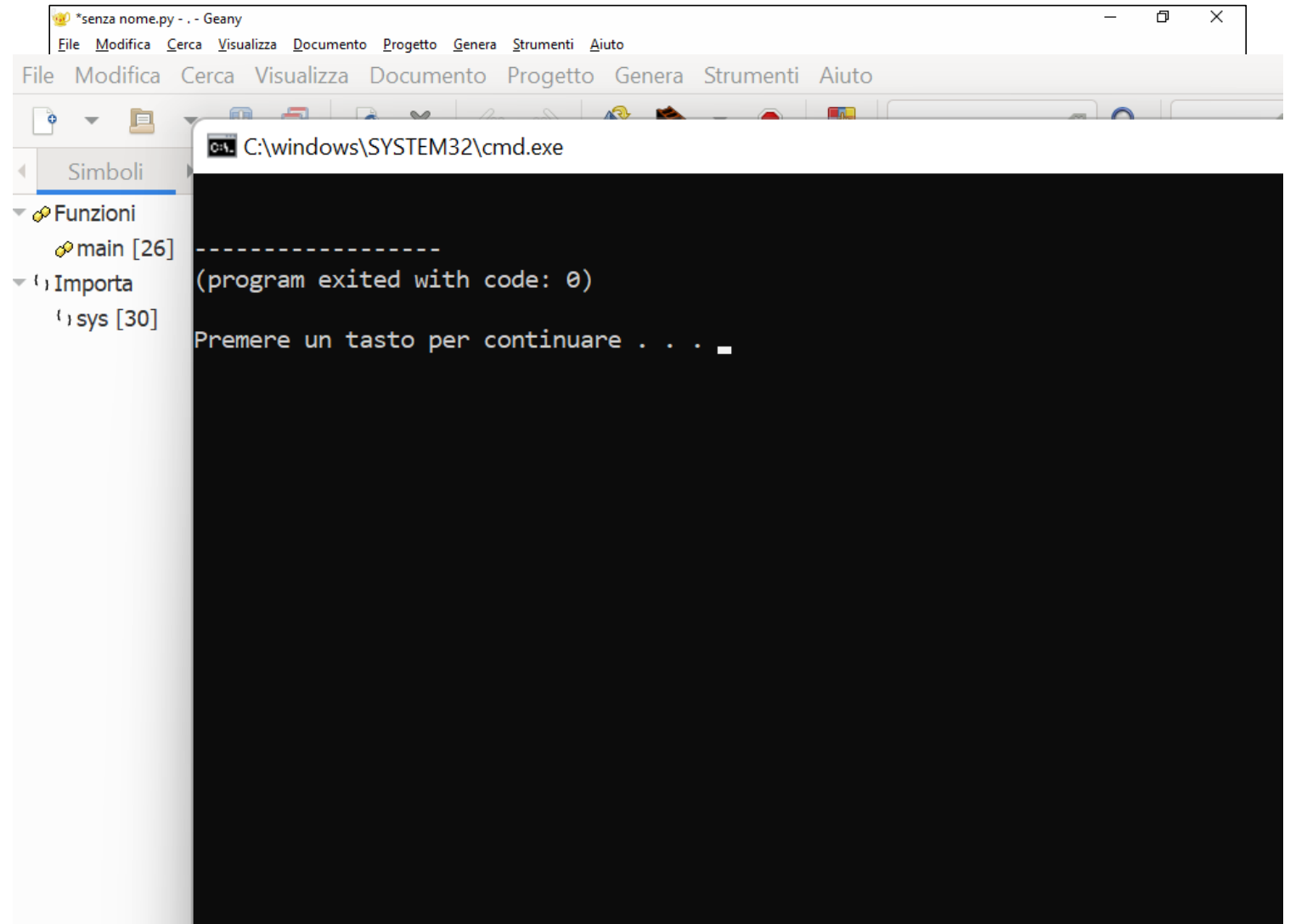
```
1 #!/usr/bin/env python
2 # -*- coding: utf-8 -*-
3 #
4 #  senza nome.py
5 #
6 #  Copyright 2017 Vito Gentile <vito.gentile@unipa.it>
7 #
8 #  This program is free software; you can redistribute it and/or modify
9 #  it under the terms of the GNU General Public License as published by
10 #  the Free Software Foundation; either version 2 of the License, or
11 #  (at your option) any later version.
12 #
13 #  This program is distributed in the hope that it will be useful,
14 #  but WITHOUT ANY WARRANTY; without even the implied warranty of
15 #  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
16 #  GNU General Public License for more details.
17 #
18 #  You should have received a copy of the GNU General Public License
19 #  along with this program; if not, write to the Free Software
20 #  Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston,
21 #  MA 02110-1301, USA.
22 #
23 #
24
25
26 def main(args):
27     return 0
28
29 if __name__ == '__main__':
30     import sys
31     sys.exit(main(sys.argv))
32
```

The status bar at the bottom shows the following information: riga: 1 / 32, colonna: 0, selezione: 0, INS, TAB, MOD, modalità: CRLF, codifica: UTF-8, tipo di file: Python, funzione corrente: sconosciuto.

Cliccando su *Genera*  
l'opzione *Execute*, dopo  
avere salvato il file, verrà  
eseguito lo script



- Ovviamente lo script creato non fa niente di particolare se non ritornare il valore di uscita 0



```
*senza nome.py - - Geany
File Modifica Cerca Visualizza Documento Progetto Genera Strumenti Aiuto
File Modifica Cerca Visualizza Documento Progetto Genera Strumenti Aiuto
Simboli
  Funzioni
    main [26]
  Importa
    sys [30]
C:\windows\SYSTEM32\cmd.exe
-----
(program exited with code: 0)
Premere un tasto per continuare . . .
```

# Ciao Mondo!

```
CiaoMondo.py - C:\Users\orazi\OneDrive\Documenti\Python_Code - Geany
File Modifica Cerca Visualizza Documento Progetto Genera Strumenti Aiuto
Simboli CiaoMondo.py x
Funzioni
  main [26]
Importa
  sys [31]
5 #
6 # Copyright 2022 orazi <orazi@TURING>
7 #
8 # This program is free software; you can redistribute it and/or modify
9 # it under the terms of the GNU General Public License as published by
10 # the Free Software Foundation; either version 2 of the License, or
11 # (at your option) any later version.
12 #
13 # This program is distributed in the hope that it will be useful,
14 # but WITHOUT ANY WARRANTY; without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
16 # GNU General Public License for more details.
17 #
18 # You should have received a copy of the GNU General Public License
19 # along with this program; if not, write to the Free Software
20 # Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston,
21 # MA 02110-1301, USA.
22 #
23 #
24
25
26 def main(args):
27     print("Ciao Mondo!")
28     return 0
29
30 if __name__ == '__main__':
31     import sys
32     sys.exit(main(sys.argv))
33
```

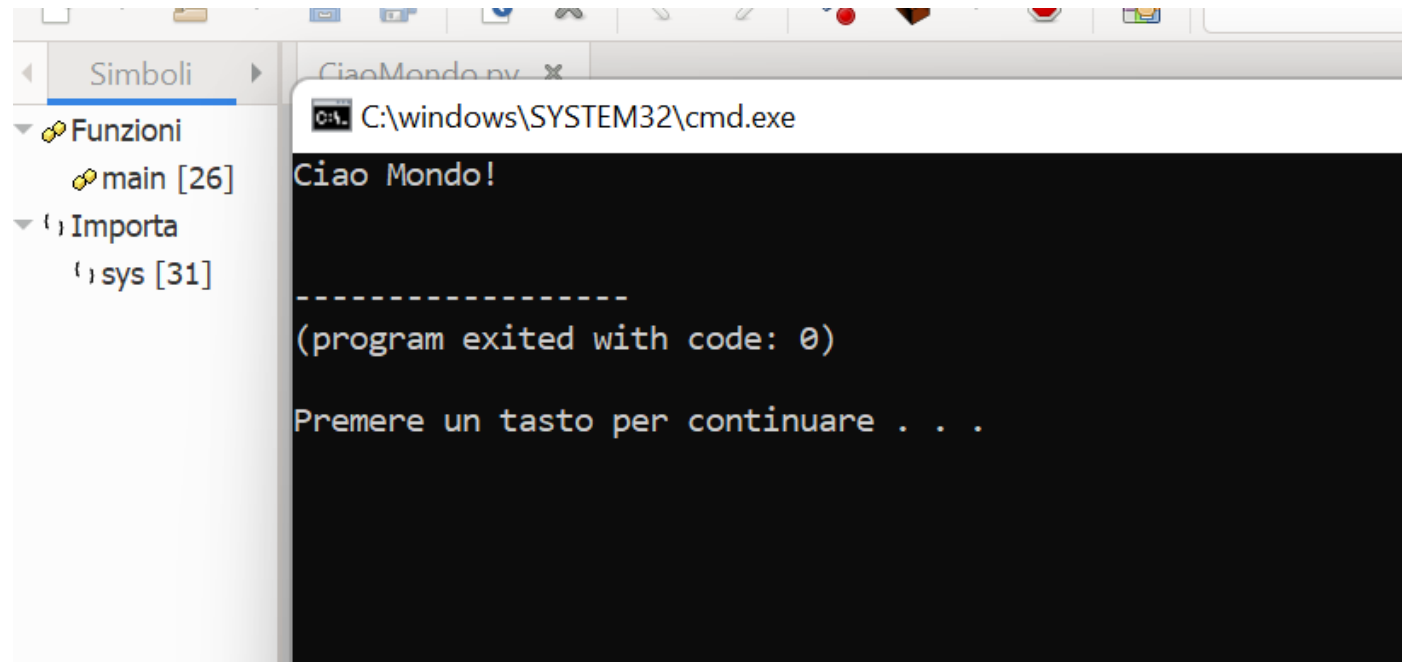
# Ciao Mondo!

```
24
25
26 def main(args):
27     print("Ciao Mondo!")
28     return 0
29
30 if __name__ == '__main__':
31     import sys
32     sys.exit(main(sys.argv))
33
```



# Ciao Mondo!

- Lo script esegue la main function che stampa "Ciao Mondo!" oltre che ritornare il valore 0



...per casa...

- Installare Python
- Installare Geany
- Sostituire, all'intero contenuto del file *.py* generato automaticamente da Geany, quanto segue:

```
a = 5  
b = 7  
print(a)  
print(b)
```

- Salvare ed eseguire questo script tramite Geany

# Approfondimenti

- Installare Python (su Windows, Linux e MacOS):
  - <http://www.html.it/pag/15609/installare-python/>
- Approfondimenti su `__name__`:
  - [https://docs.python.org/3/library/\\_\\_main\\_\\_.html](https://docs.python.org/3/library/__main__.html)
- Installazione e uso (avanzato) di Python:
  - <https://gist.github.com/ricpol/2ca0ae46f02bfddf08036fa85519aa97>