

PYTHON LESSON 2



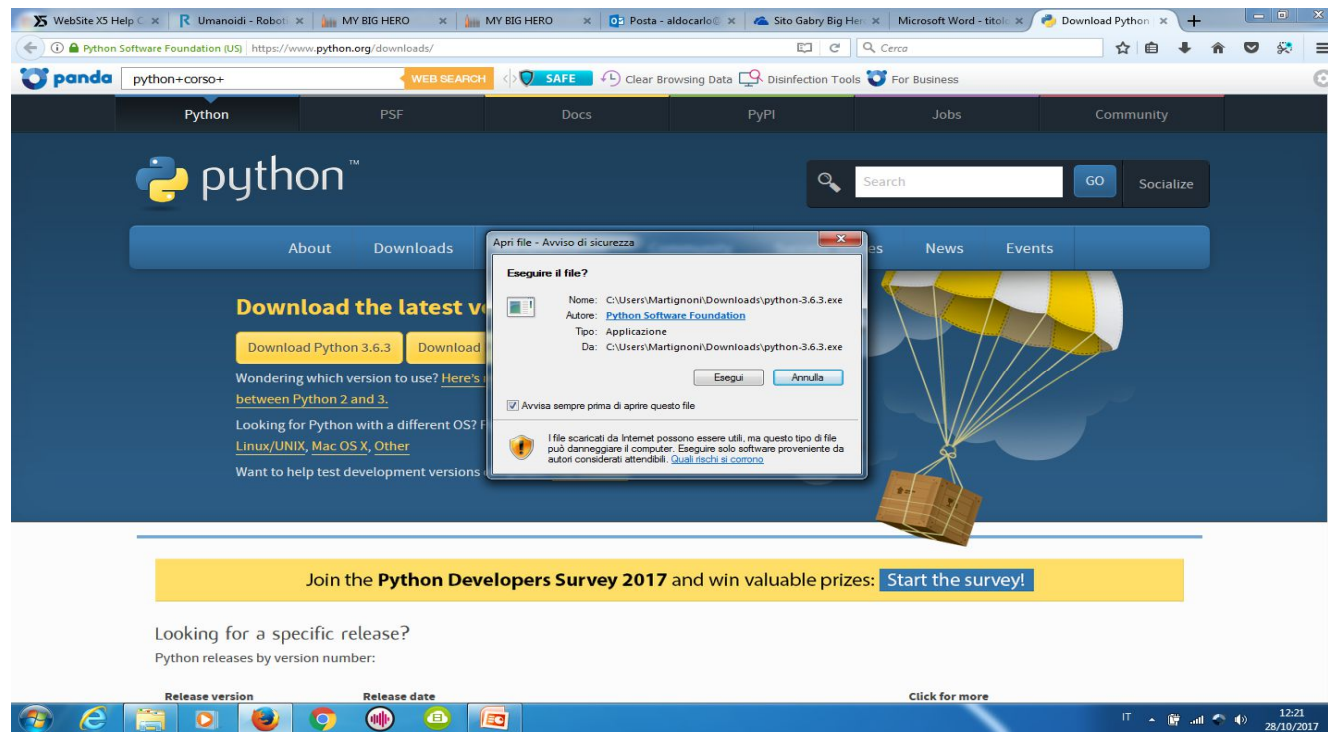
python

PYTHON PRELIMINARI

Prima cosa da fare è scaricare il programma dal seguente link


<http://www.python.org/download/>

Se tutto va come si deve avrete una maschera di questo tipo



PYTHON PRELIMINARI

Ora lanciate il programma è attendete istallazione una volta istallato siamo pronti per iniziare . Vedremo alcuni comandi base e come utilizzarli.
Ora lanciamo la shell



```
Python 3.6.5 Shell
File Edit Shell Debug Options Window Help
Python 3.6.5 (v3.6.5:f59c0932b4, Mar 28 2018, 16:07:46) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> |
```

Ln: 3 Col: 4

PYTHON PRELIMINARI

Ora possiamo vedere lo strumento per la programmazione che Python mette a disposizione, denominato IDLE, esso consente, tramite una interfaccia grafica di lanciare istruzioni basate sul linguaggio e creare file eseguibili con estensione ".py".

Avviato IDLE si potrà lanciare immediatamente un primo comando basato su Python per cominciare ad analizzarne la sintassi di base; le istruzioni dovranno essere scritte dopo i simboli ">>>" che indicano la modalità interattiva, cioè la possibilità di comunicare con l'interprete (si tratta di un'interfaccia del tutto simile alla riga di comando del prompt di Windows).

Una delle cose più importanti nella scrittura di un programma in Python è "Indentazione"

CHE COSA E' INDENTAZIONE

A differenza di altri linguaggi che delimitano blocchi di codice con parentesi grafe (come C, C++ e Java) o con parole riservate come begin/end, Python usa l'indentazione. Indentare il codice è una pratica comune in tutti i linguaggi, perché semplifica la lettura del codice e la compressione della sua struttura. Anzichè usare due meccanismi separati per compilatori/interpreti (parentesi o keyword) e persone (indentazione), per Python si è scelto di usare l'indentazione per entrambi. Questo significa che, in Python, l'indentazione è significativa, e che indentare in modo incorretto può portare a comportamenti sbagliati del programma o a errori.

A causa della natura degli editor di testo sui sistemi non UNIX non è consigliato utilizzare un misto di spazi e tab per indentare una singola riga di codice sorgente. Si deve anche notare che differenti piattaforme possono limitare il numero massimo di livelli di indentazione.

PYTHON I COMMENTI

Inserire i commenti su un codice è fondamentale per la comprensione dello stesso Python supporta i commenti, cioè delle annotazioni che lo sviluppatore potrà inserire all'interno del sorgente per incrementarne il livello di leggibilità; grazie a dei caratteri appositamente concepiti per la loro delimitazione, l'interprete del linguaggio riconoscerà i commenti come tali evitando che essi entrino nel flusso di esecuzione del codice e siano visibili all'interno degli output.

E' possibile distinguere tra commenti a linea singola e commenti multilinea; i commenti a linea singola vengono ospitati su una sola riga e introdotti tramite un delimitatore iniziale che è il carattere cancelletto (#):

```
>>> #Questo è un commento su singola linea.
```

commenti multilinea, cioè quelli che si estendono su più righe, vengono invece delimitati tramite tre apici singoli posti sia in apertura che in chiusura del commento:

```
>>>''' Questo è un commento multilinea. '''
```

PYTHON I COMMENTI

Inserire i commenti su un codice è fondamentale per la comprensione dello stesso Python supporta i commenti, cioè delle annotazioni che lo sviluppatore potrà inserire all'interno del sorgente per incrementarne il livello di leggibilità; grazie a dei caratteri appositamente concepiti per la loro delimitazione.

E' possibile distinguere tra commenti a linea singola e commenti multilinea; i commenti a linea singola vengono ospitati su una sola riga e introdotti tramite un delimitatore iniziale che è il carattere cancelletto (#):

```
>>> #Questo è un commento su singola linea.
```

commenti multilinea, cioè quelli che si estendono su più righe, vengono invece delimitati tramite tre apici singoli (""") posti sia in apertura che in chiusura del commento:

```
>>>""" Questo è un  
commento multilinea. """
```

PYTHON I COMMENTI

Questo ultimo formato viene sconsigliato dagli sviluppatori di Python in quanto potrebbe generare confusione nella lettura del codice; inoltre, nelle più recenti versioni di Python l'uso della delimitazione `''' ... '''` può essere interpretata come istruzione per l'output della stringa contenuta tra gli apici.

Pertanto, è meglio l'utilizzo dei commenti linea singola anche per il multilinea, come nell'esempio seguente:

```
>>> # Questo  
#è un commento  
  
#multilinea.
```