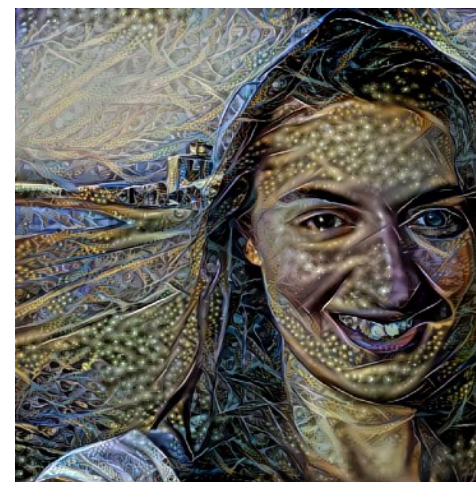
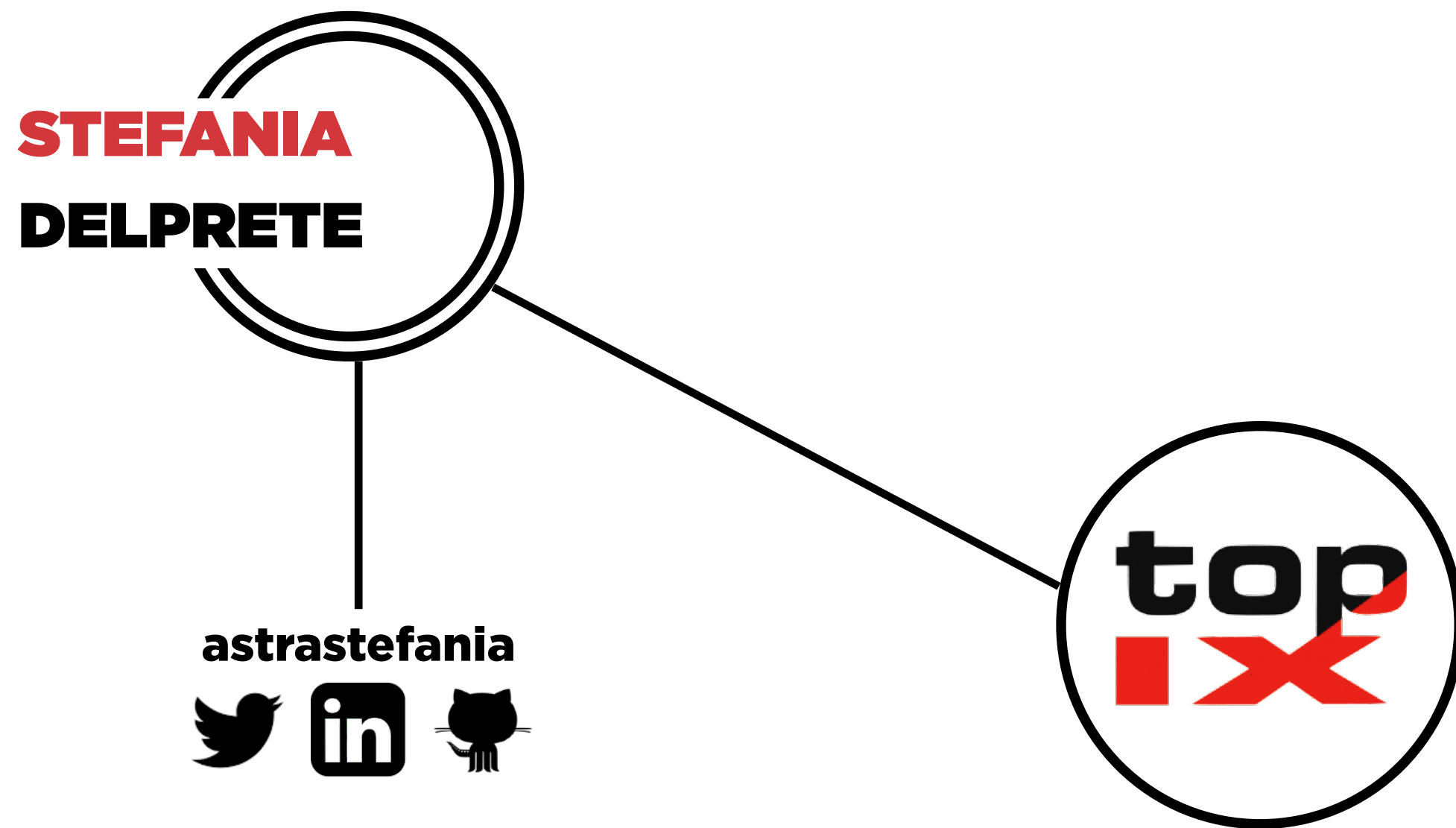


# INTRODUZIONE A PYTHON

**CORSO ENGIM IFTS  
MODULO PYTHON**



# **AGENDA**

- i. Cos'è Python?**
- ii. Python interpreter e script da terminale**
- iii. Anaconda: Spyder e Jupyter Notebook**

**COS'È PYTHON?**

# **CHE COS'È PYTHON?**

**/ Python è stato creato da Guido von Rossum nei Paesi Bassi e rilasciato nel 1991**

**Linguaggio di programmazione:**

**/ Interpretato**

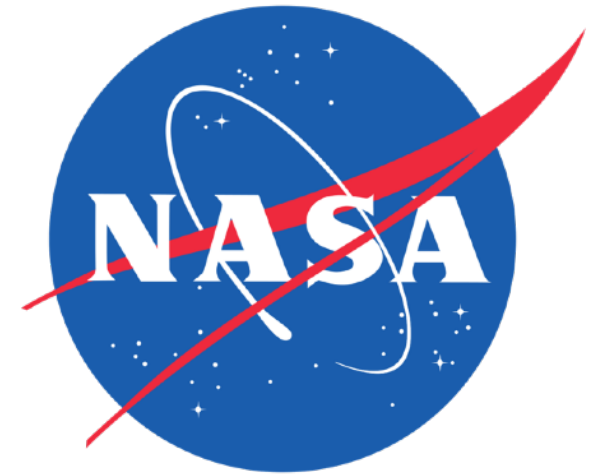
**/ Interattivo**

**/ Orientato agli oggetti**

# **PUNTI DI FORZA DI PYTHON**

- / Leggibilità del codice**
- / Sintassi che permette di rappresentare concetti in poche righe**
- / Versatibilità di utilizzo**

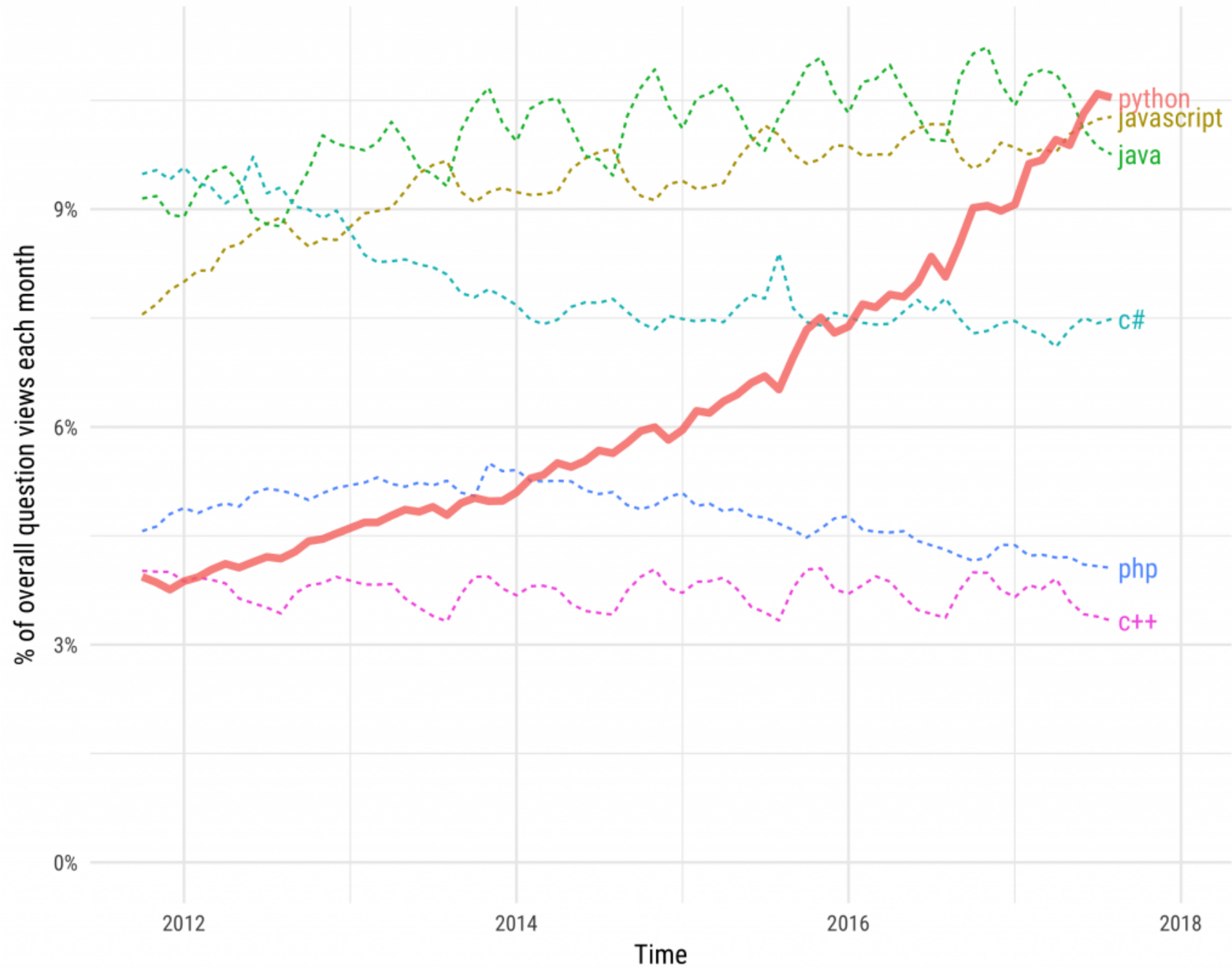
# CHI STA USANDO PYTHON?



# PYTHON SU STACKOVERFLOW

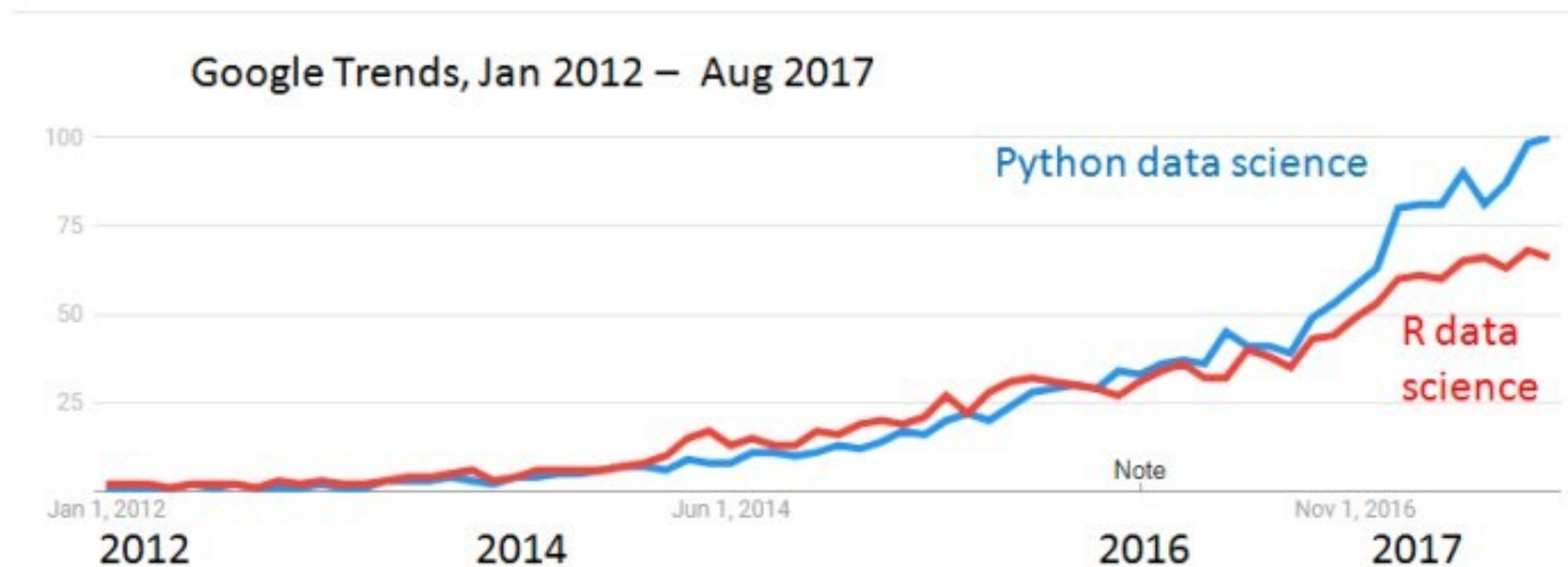
## Growth of major programming languages

Based on Stack Overflow question views in World Bank high-income countries

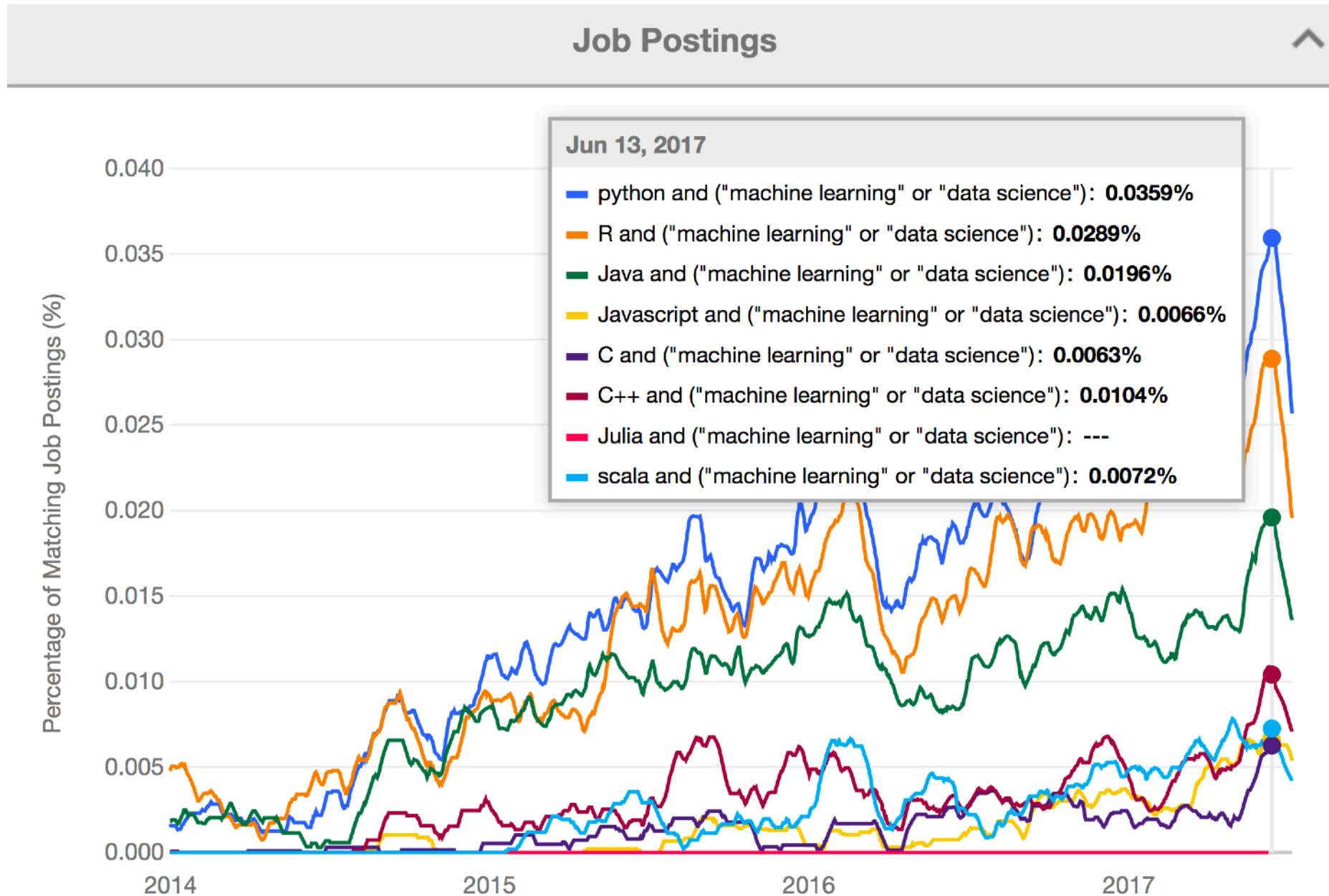




# PYTHON E RICERCHE SU DATA SCIENCE

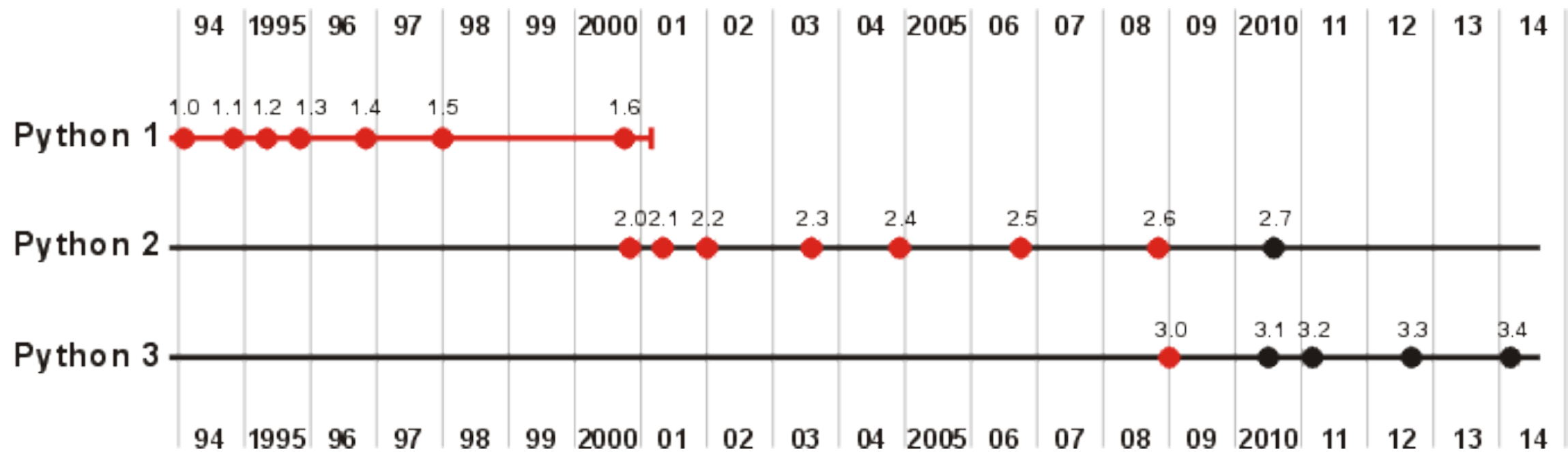


# PYTHON E OFFERTE DI LAVORO



<https://www.indeed.com/jobtrends/q-python-and-%28%22machine-learning%22-or-%22data-science%22%29-q-R-and-%28%22machine-learning%22-or-%22data-science%22%29-q-Java-and-%28%22machine-learning%22-or-%22data-science%22%29-q-Javascript-and-%28%22machine-learning%22-or-%22data-science%22%29-q-C-and-%28%22machine-learning%22-or-%22data-science%22%29-q-C++-and-%28%22machine-learning%22-or-%22data-science%22%29-q-Julia-and-%28%22machine-learning%22-or-%22data-science%22%29-q-scala-and-%28%22machine-learning%22-or-%22data-science%22%29.html>

# VERSIONI DI PYTHON



**/ Python 2.7 non sarà più mantenuto dopo il 2020**

# **PYTHON INTERPRETER**

# **PYTHON INTERPRETER**

**Da terminale possiamo verificare la versione di Python:**

```
$ python -V
```

**Per accedere al Python interpreter (con la versione di default):**

```
$ python
```

# **PYTHON INTERPRETER**

**Per forzare una versione:**

```
$ python3.6
```

**Per uscire dal Python interpreter:**

```
CTRL+D
```

# PYTHON INTERPRETER

## Prime prove in Python

```
>>> 3 + 4
```

```
>>> print("Hello World")
```

```
>>> nome = 'Stefania'
```

```
>>> print("Ciao, " + nome + "!")
```

# ZEN OF PYTHON

```
>>> import this
```

The Zen of Python, by Tim Peters

Beautiful is better than ugly.

Explicit is better than implicit.

Simple is better than complex.

Complex is better than complicated.

Flat is better than nested.

Sparse is better than dense.

Readability counts.

Special cases aren't special enough to break the rules.

Although practicality beats purity.

Errors should never pass silently.

Unless explicitly silenced.

In the face of ambiguity, refuse the temptation to guess.

There should be one-- and preferably only one --obvious way to do it.

Although that way may not be obvious at first unless you're Dutch.

Now is better than never.

Although never is often better than \*right\* now.

If the implementation is hard to explain, it's a bad idea.

If the implementation is easy to explain, it may be a good idea.

Namespaces are one honking great idea -- let's do more of those!



# **PEP 8 STILE OLTRE L'IDENTTAZIONE**

**PEP** (Python Enhancement Proposals) sono guide ufficiali in cui vengono discusse e proposte dalla comunità la caratteriale del linguaggio.

**PEP 8** contiene le linee guida e convenzioni per scrivere in Python in modo più leggibile possibile.

**<https://www.python.org/dev/peps/pep-0008>**

**SCRIPT DA TERMINALE**

# **ESPLORAZIONI NEL TERMINALE**

**Dove sei in questo momento?**

```
$ pwd
```

**Cosa c'è nella cartella in cui siamo?**

```
$ ls
```

**Spostiamoci nella home:**

```
$ cd
```

**Spostiamoci su Scrivania:**

```
$ cd Scrivania
```

# **ESPLORAZIONI NEL TERMINALE**

**Creiamo una cartella esempi:**

```
$ mkdir esempi
```

**E spostiamoci dentro:**

```
$ cd esempi
```

**Creiamo ed editiamo il file ciao.py:**

```
$ nano ciao.py
```

**‘nano’ è uno dei numerosi editor disponibili!**

# **CREIAMO E LANCIAMO LO SCRIPT**

## **Dentro l'editor 'nano' scriviamo:**

```
nome = 'Stefania'  
print("Ciao, " + nome + "!")  
print(nome*3)
```

## **Salviamo e usciamo dall'editor:**

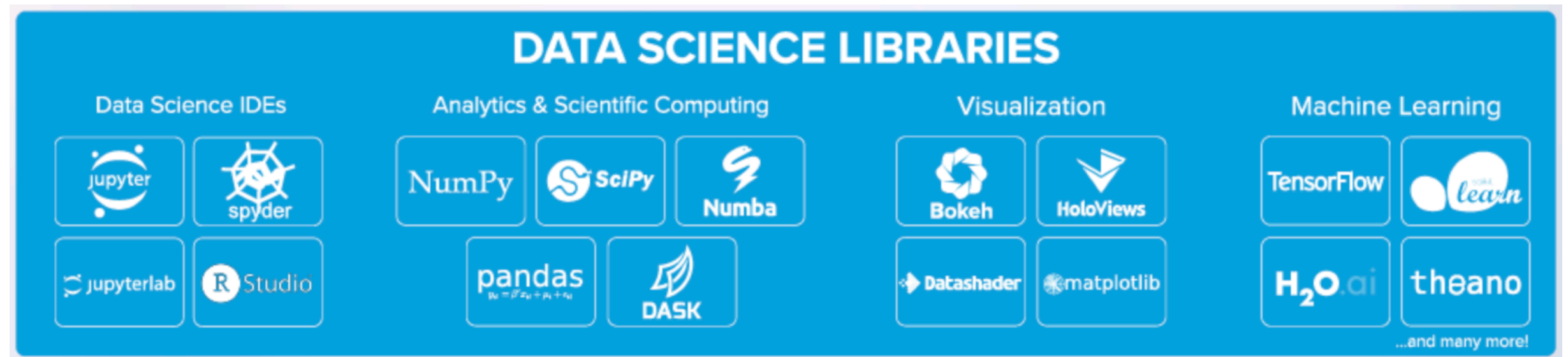
CTRL+X, Y o S, invio

## **Lanciamo lo script:**

```
$ python ciao.py
```


# **ANACONDA: SPYDER E JUPYER NOTEBOOK**

# ANACONDA



**Anaconda Distribution** contiene le più usate librerie Python per data science e machine learning e IDE (Integrated Development Environment) accessibili via terminale o interfaccia grafica.

# ANACONDA INTERFACCIA GRAFICA

 **ANACONDA** NAVIGATOR

Sign in to Anaconda Cloud

Home

Environments

Projects (beta)




Learning

Community


Documentation


Developer Blog


Feedback


  

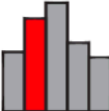
Applications on anaconda3 Channels Refresh


  
jupyterlab  
0.27.0  
An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.  
Launch


  
notebook  
5.0.0  
Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.  
Launch

  
qtconsole  
4.3.1  
PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.  
Launch

  
spyder  
3.2.4  
Scientific PYTHON Development Environment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features  
Launch

  
glueviz  
0.12.0  
Multidimensional data visualization across files. Explore relationships within and among related datasets.  
Install

  
orange3  
3.4.1  
Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.  
Install

  
rstudio  
1.1.383  
A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.  
Install

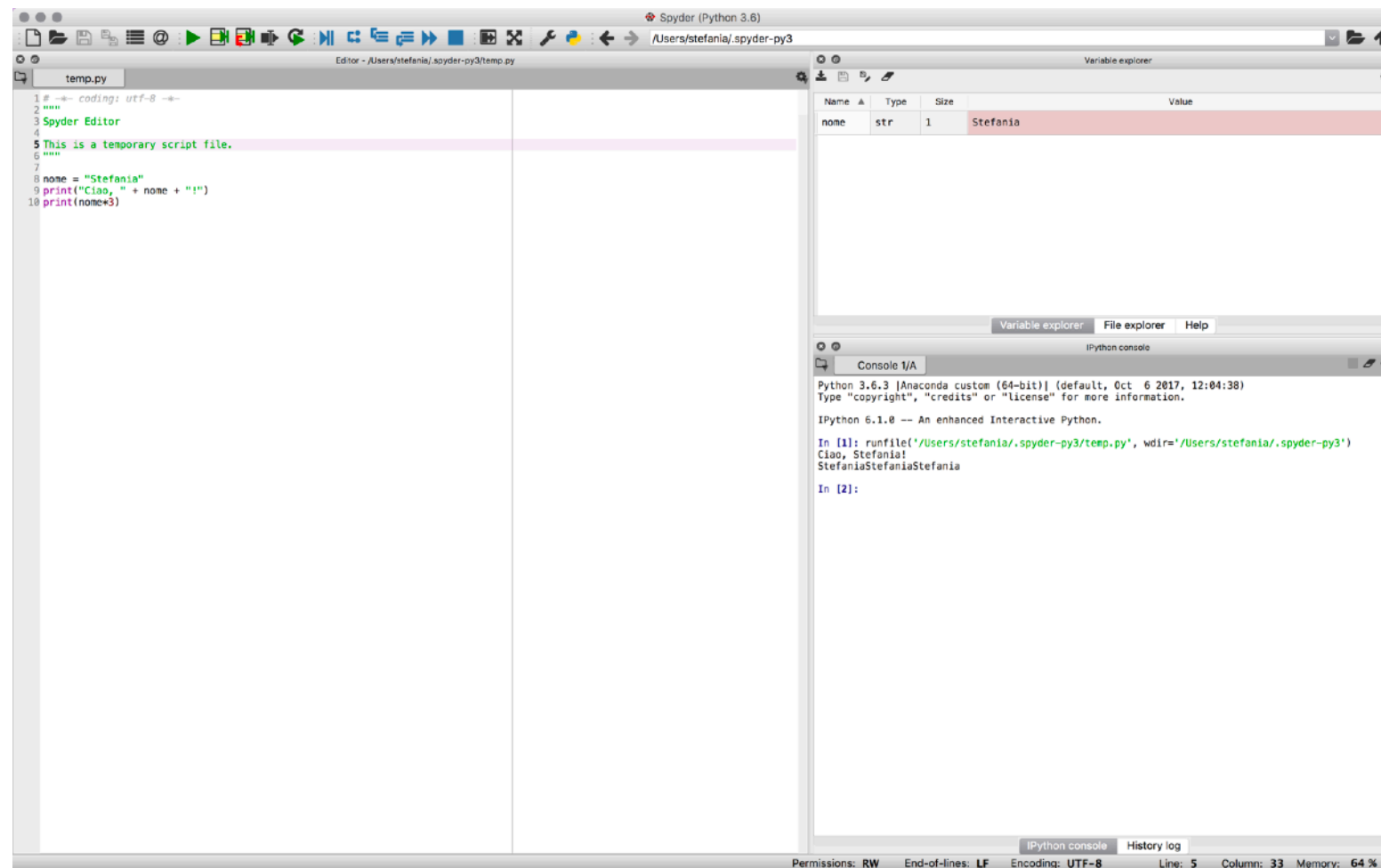


# SPYDER

**Lanciamo Spyder da terminale:**

**\$ spyder**

**Creiamo e lanciamo lo script di prima**



# JUPYTER NOTEBOOK

**Lanciamo Jupyter Notebook da terminale:**

```
$ jupyter notebook
```

**Creiamo lo script di prima una riga alla volta e procediamo insieme!**

**RISORSE**

# RISORSE **PYTHON**



**/ Punto di partenza**  
**<https://www.python.org>**

**/ Documentazione online**  
**<https://docs.python.org/3>**

**/ Discussioni su StackOverflow**  
**<https://stackoverflow.com/questions/tagged/python>**

**/ Codice di stile**  
**<https://www.python.org/dev/peps/pep-0008>**

**/ Zen of Python**  
**<https://www.python.org/dev/peps/pep-0020>**

# RISORSE **JUPYTER NOTEBOOK**

**/ Punto di partenza**  
**<http://jupyter.org>**

**/ Jupiter online (non sempre disponibile)**  
**<https://try.jupyter.org>**

**/ Galleria su Anaconda**  
**<https://anaconda.org/gallery>**



# EVENTI PYTHON E DATA SCIENCE

**/ PyCon Italia**  
**<https://www.pycon.it>**

**/ Eventi PyData nel mondo**  
**<https://pydata.org>**



**stefania.delprete@top-ix.org**

**www.top-ix.org**

**@top\_ix**

**GRAZIE!**