Programmazione 1

Introduzione al corso

Stefano Gualandi, @famo2spaghi
Studio E14, Dipartimento di Matematica

Blog: http://stegua.github.io

Due questionari

1. Raccolta dati anagrafici per creazione account nei laboratori di ingegneria

2. Questionario conoscitivo perchè capire le vostre conoscenze in ambito informatico

3. Andare all'indrizzo:

http://mate.unipv.it/gualandi/programming/

Informazioni generali

- **Siti del corso**

  http://matematica.unipv.it/gualandi/programming
  https://github.com/mathcoding/programming

- **Orario lezioni**:
  - Lunedi 14h00/16h00 – Aula B2
  - Mercoledi 14h00/15h00 – Laboratorio didattico
  - Venerdi 11h00/13h00 – Aula C3 o Laboratorio didattico

- **Tutorato**: sarà attivato a partire da metà ottobre

Informazioni generali

- **Esame**:
  - Prova di programmazione in aula informatica
  - Durata 2 ore
  - A breve, saranno online i testi degli esercizi d'esame dell'anno precedente (2017-18)

- **Ricevimento**:
  - In qualsiasi momento potete richiedere ricevimento via email (usate l'email **@universitadipavia.it!**)
  - Appena possibile stabilisco un giorno e orario in cui non avete lezione

Comunicazioni con il docente

- Usare sempre l'email di ateneo
  nome.cognome@**universitadipavia.it**

- Cercare di essere precisi nel fare le domande

- Non avere timore di venire a ricevimento

Testi di riferimento

Slides e script usati a lezione
(in italiano), reperibili sul sito

Materiale didattico

Gli script Python usati a lezioni sono continuamente aggiornati su GitHub al repository seguente: Programmazione 1

Lucidi usati a lezione (per i notebooks python consultare il sito su github):

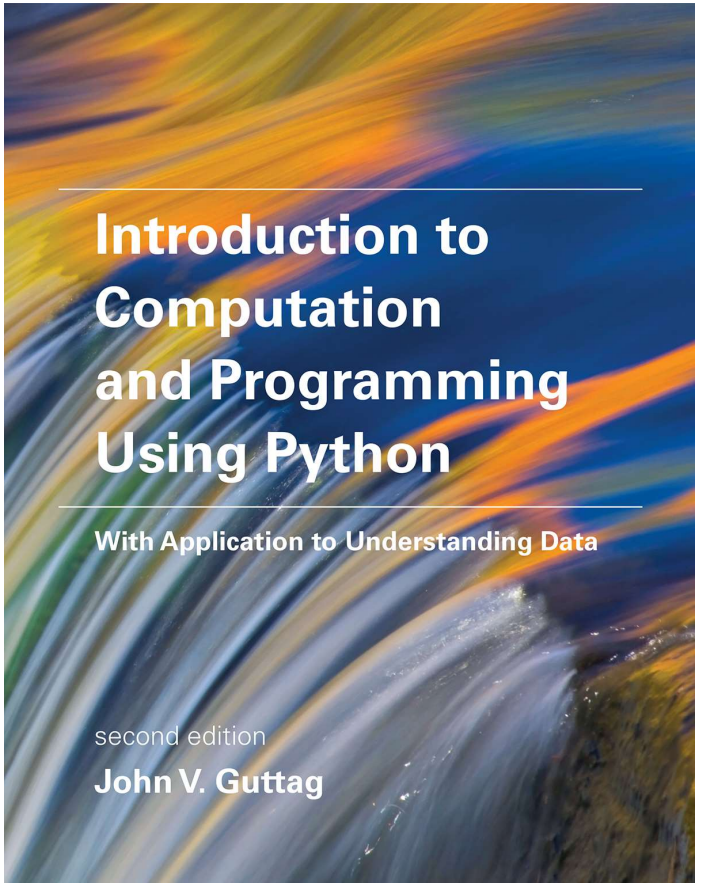
- Lucidi di introduzione al corso ...

Letture Consigliate

Links Utili

- Versione di Python consigliata: Anaconda
- Applicazione per connettersi a GitHub: GitHub Desktop
- GitHub Student Pack
- Documentazione ufficiale di Python 3.x: docs

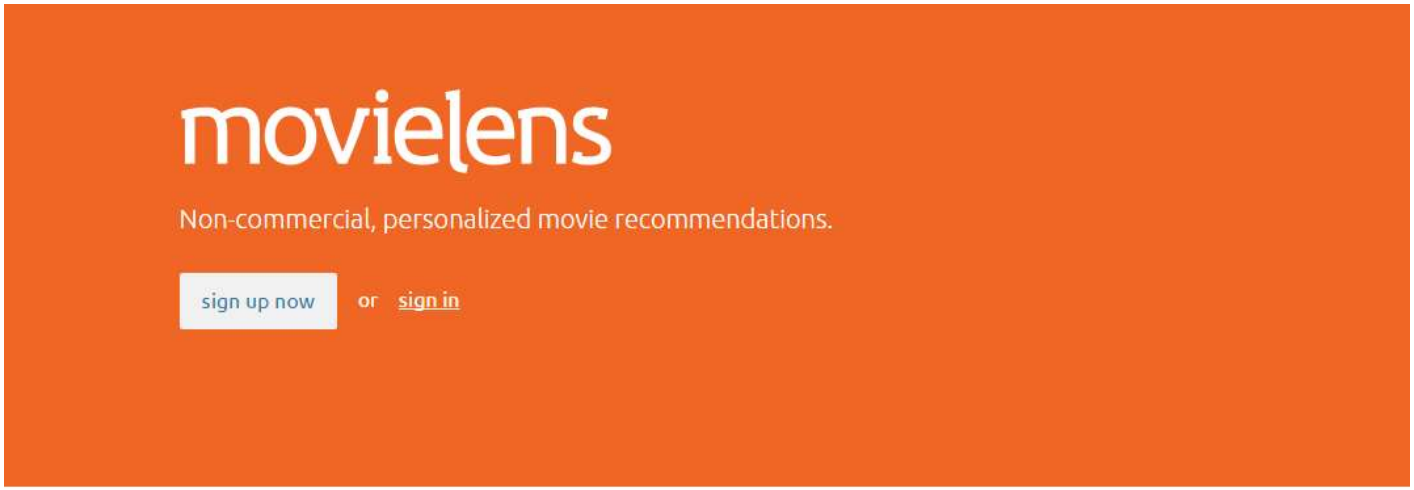
Testo di riferimento (in inglese)



Obiettivi del corso

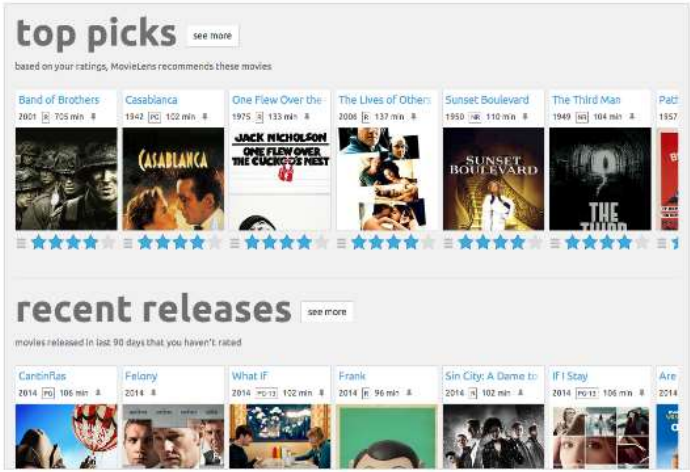
1. Introdurre i concetti fondamentali di programmazione
2. Insegnare ad usare la programmazione come supporto alla risoluzione di problemi
3. Insegnare un linguaggio di programmazione multiparadigma (Python)
4. Stimolare la vostra curiosità e fantasia
5. Stimolare la vostra curiosità e fantasia
6. Stimolare la vostra curiosità e fantasia
7. ***Coding is fun!***

http://movielens.org



recommendations

MovieLens helps you find movies you will like. Rate movies to build a custom taste profile, then MovieLens recommends other movies for you to watch.



Programma del corso

- Introduzione al linguaggio di programmazione Python
- Elementi di Programmazione funzionale
- Elementi di Programmazione procedurale
- Elementi di Programmazione orientata agli oggetti
- Cenni di complessità algoritmica
- Algoritmi di ricerca e di ordinamento
- Algoritmi di programmazione dinamica

Risoluzione di problemi

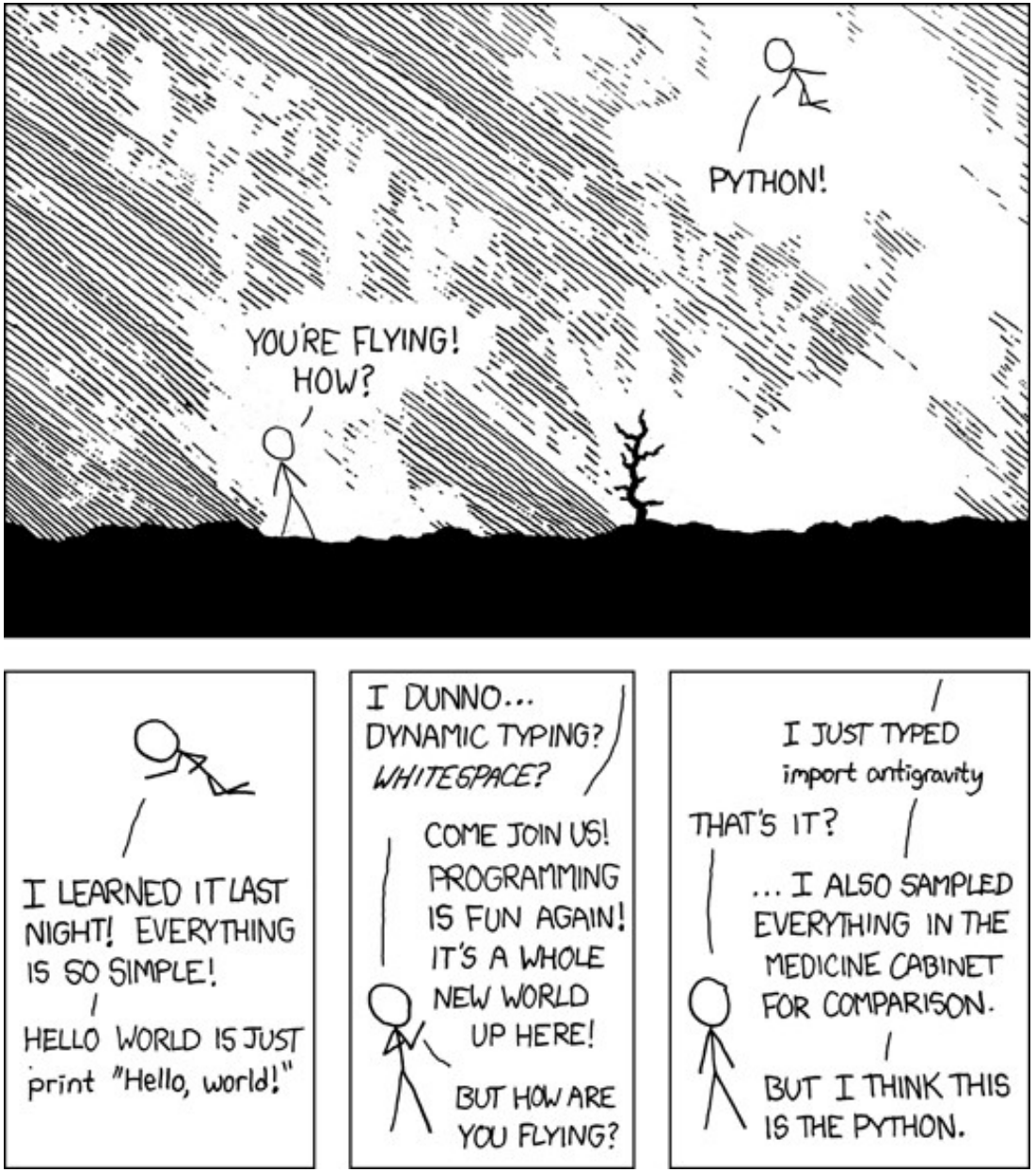
- Formalismo per rappresentare un problema (e.g. grafi)
- Definizione formale del problema e degli obiettivi (e.g., quale funzione obiettivo?)
- Soluzione ammissibile, soluzione ottima
- Problema risolvibile/non risolvibile al calcolatore (complessità computazionale)
- Tempo di esecuzione vs. tempo di sviluppo

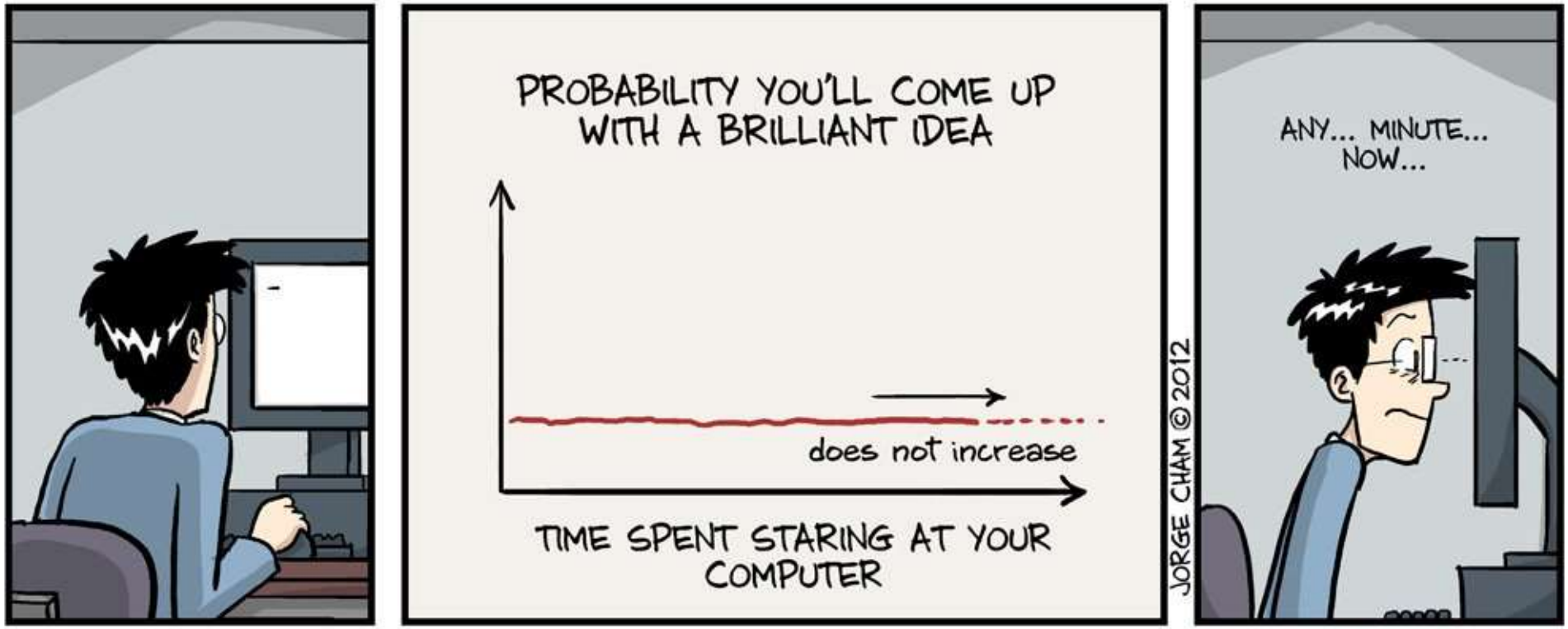
Linguaggi di progammazione

- Perchè Python?
  - Semplice e divertente!

- Alternative:
  - C++ - http://en.cppreference.com
  - Julia - http://julialang.org/
  - Haskell - http://www.haskell.org/
  - R - http://www.r-project.org/
  - C# - **google-it-yourself**
  - Java - http://www.java.com

import antigravity




PROBABILITY YOU'LL COME UP WITH A BRILLIANT IDEA
does not increase
TIME SPENT STARING AT YOUR COMPUTER
ANY... MINUTE... NOW...
JORGE CHAM © 2012
WWW.PHDCOMICS.COM

Perchè Python?

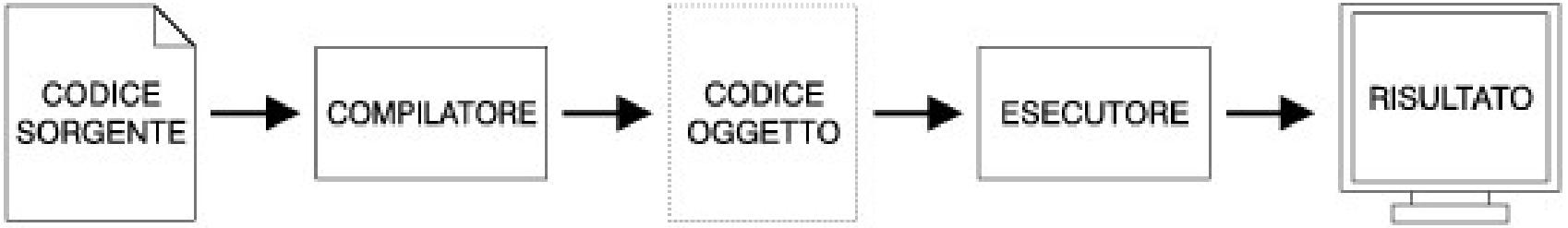
PRO:

- È interattivo (interpretato, non compilato)
- È semplice (no gestione diretta della memoria)
- È semplice *programmare* delle visualizzazioni
- È un linguaggio multiparadigma
- Ha un elevato numero di librerie semplici da installare

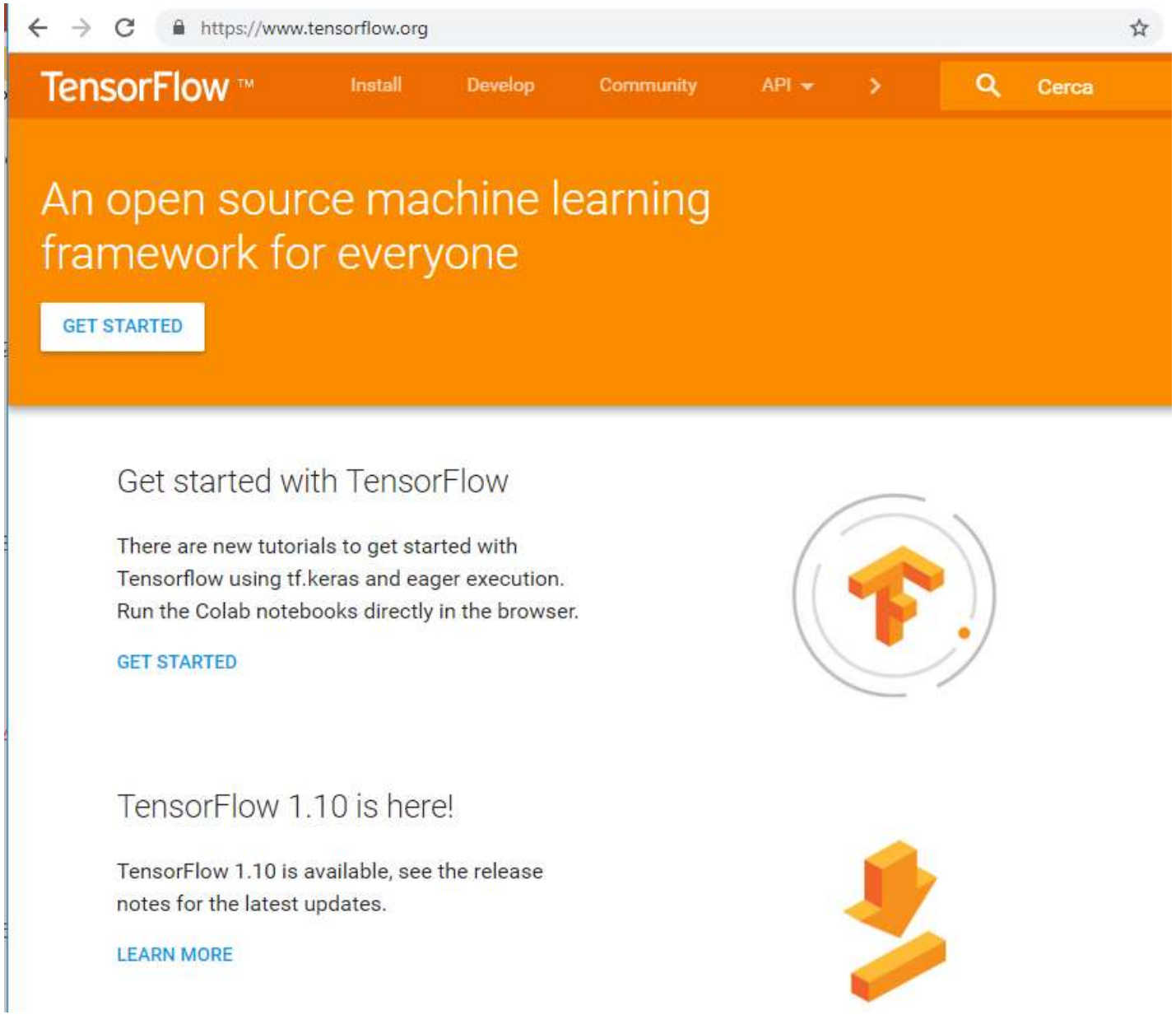
CONTRO:

1. Non é efficiente come un linguaggio compilato
2. L'identazione del codice fa parte della semantica
3. L'identazione del codice fa parte della semantica
4. L'identazione del codice fa parte della semantica

Linguaggio Interpretato vs. Compilato



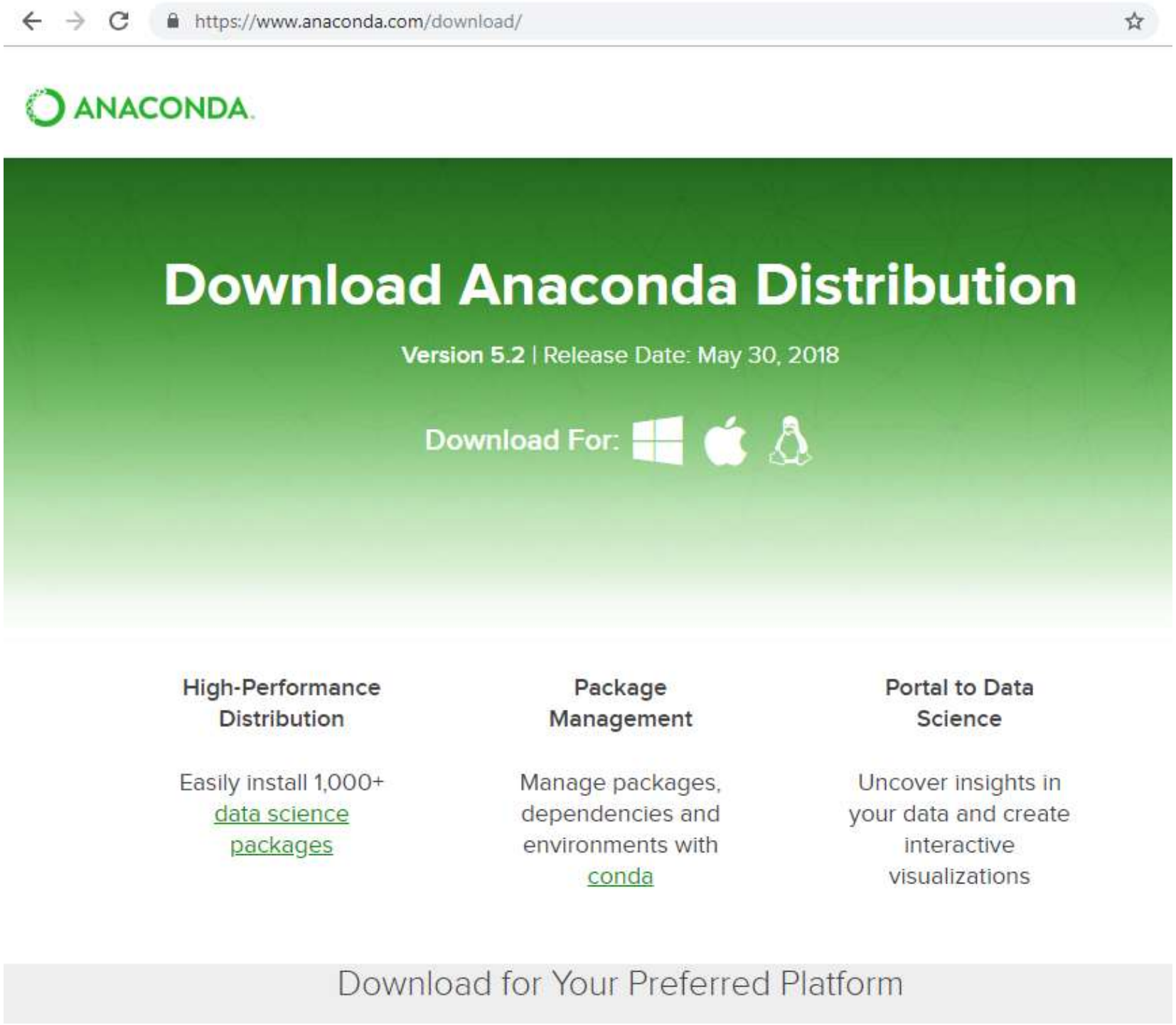
Machine Learning (at Google)



Ambiente di lavoro Python

- A lezione e in laboratorio si farà riferimento a Python >= 3.4 (**attenzione a non scaricare o usare la versione di Python 2.7**)
- Ambiente di riferimento Anaconda/Python >= 3.4:
  https://www.continuum.io/downloads

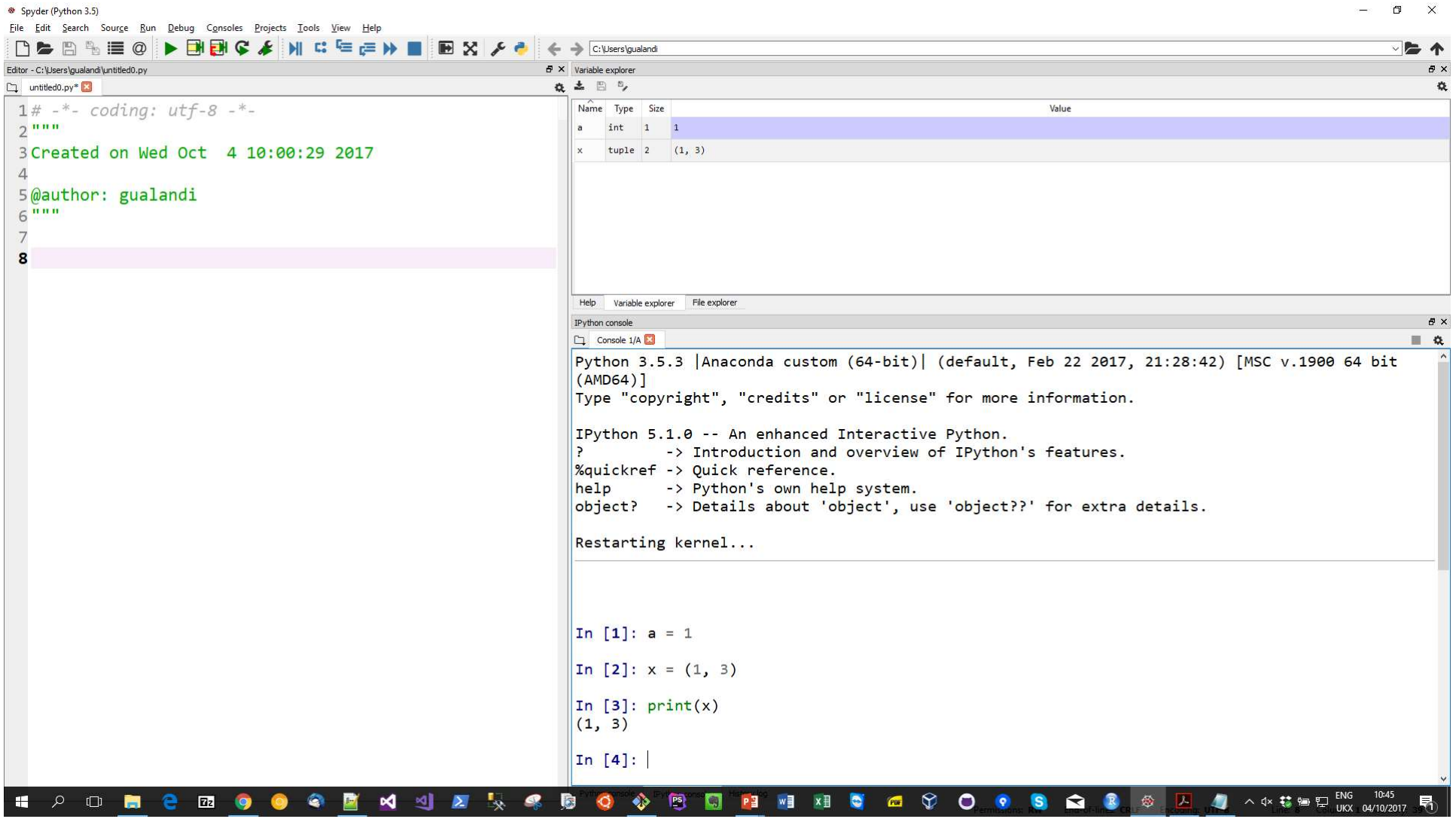
- Editors:
  - **CONSIGLIATO: Spyder3** (installato con Anaconda)
  - Vim o Emacs
  - Visual Studio (windows) o Visual Code
  - Sublime Text - http://www.sublimetext.com/
  - Rodeo - http://www.yhat.com/products/rodeo


https://www.anaconda.com/download/
ANACONDA.
Download Anaconda Distribution
Version 5.2 | Release Date: May 30, 2018
Download For:
High-Performance Distribution
Easily install 1,000+ data science packages
Package Management
Manage packages, dependencies and environments with conda
Portal to Data Science
Uncover insights in your data and create interactive visualizations
Download for Your Preferred Platform

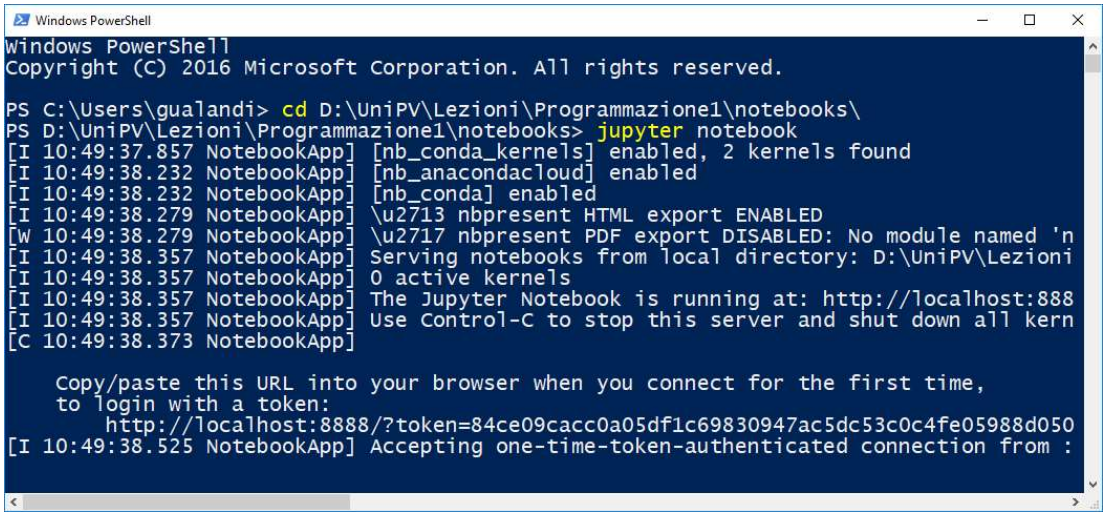
Per provare Python rapidamente

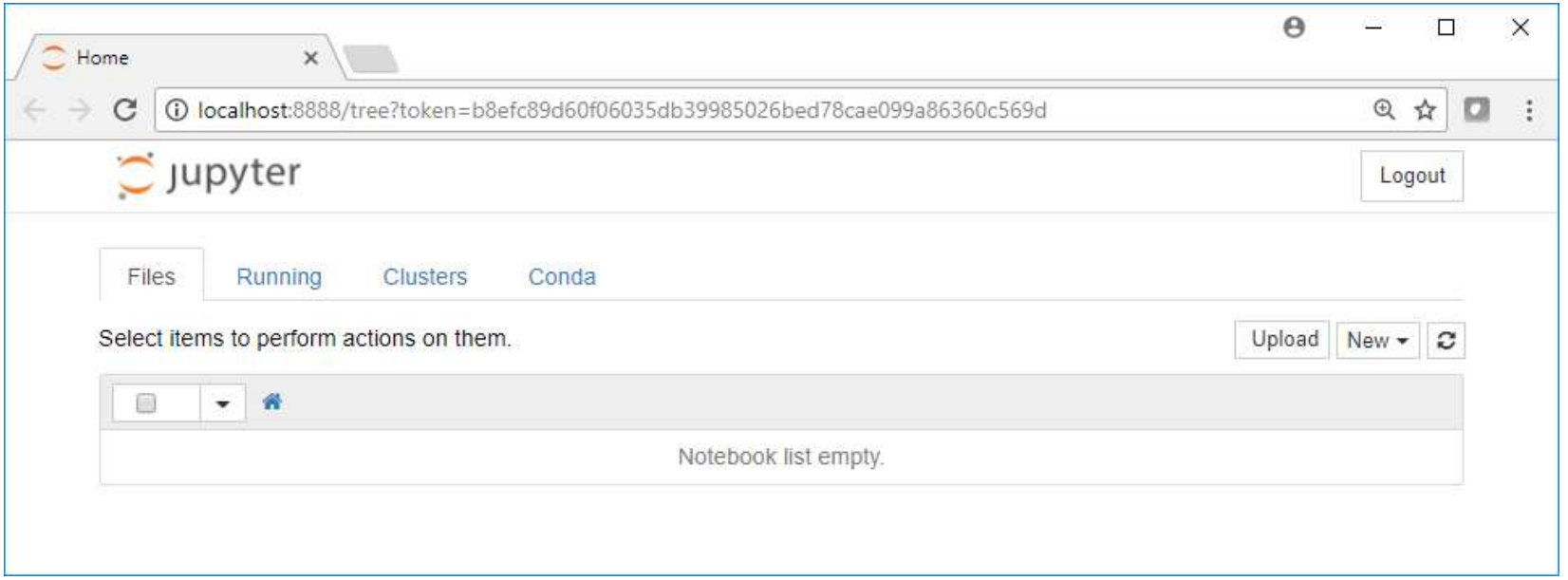
https://jupyter.org/try

“E' l'istess...“ *Spyder3 1/3*

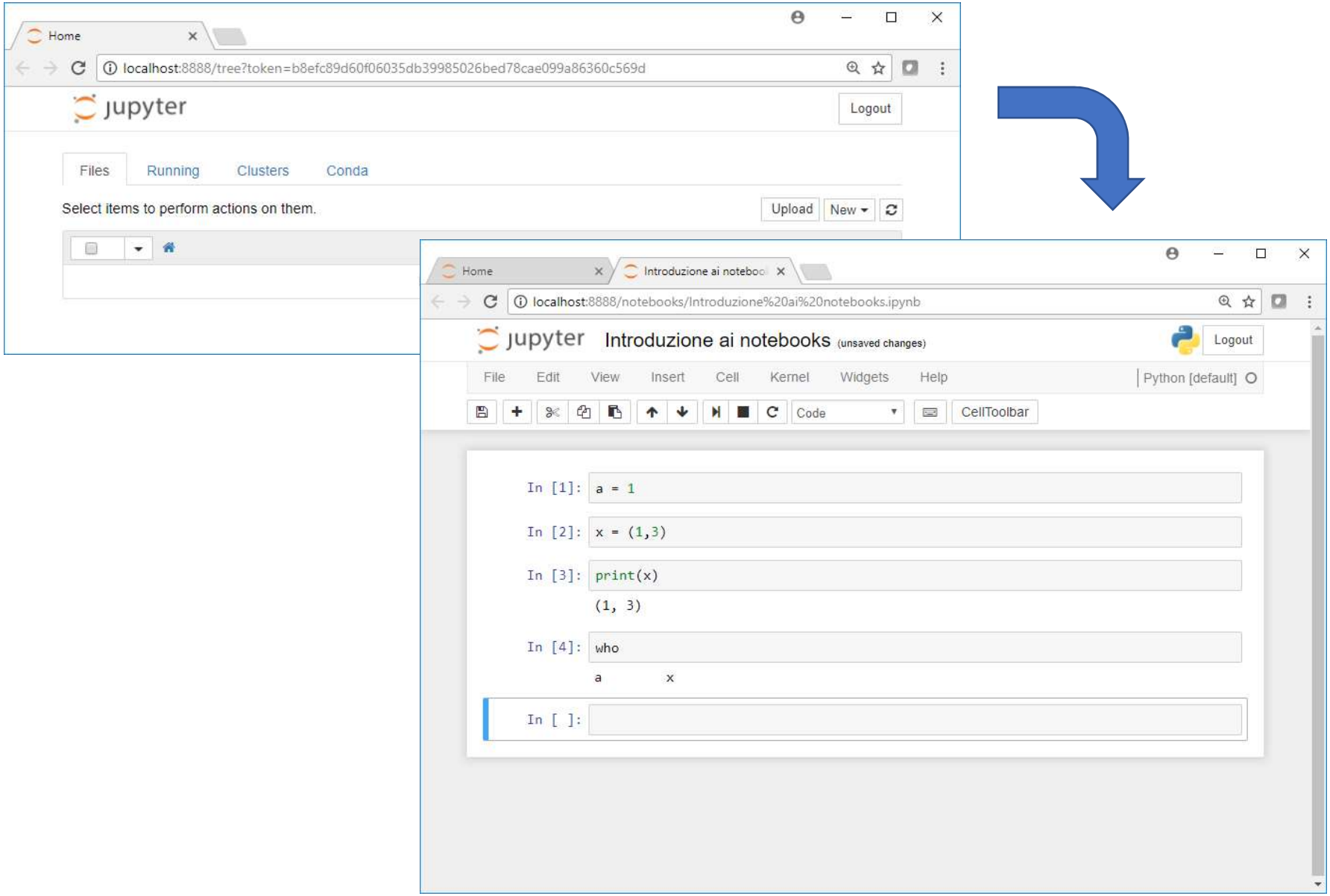


“E' l'istess...“ *Notebooks 2/3*

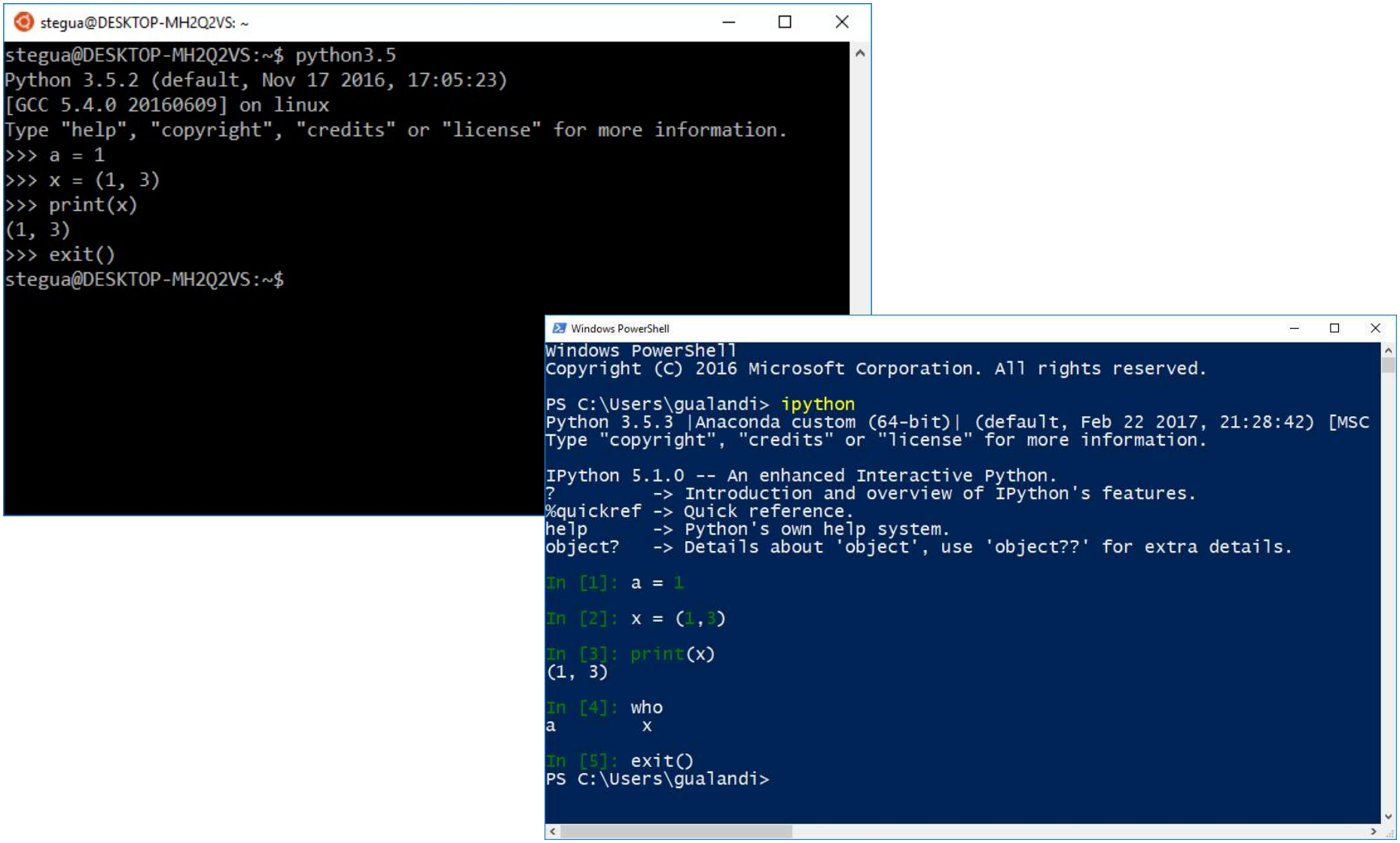




“E' l'istess...“


Home
localhost:8888/tree?token=b8efc89d60f06035db39985026bed78cae099a86360c569d
jupyter
Logout
Files
Running
Clusters
Conda
Select items to perform actions on them.
Upload
New
Home
Introduzione ai notebo
localhost:8888/notebooks/Introduzione%20ai%20notebooks.ipynb
jupyter
Introduzione ai notebooks
(unsaved changes)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Python [default]
Code
CellToolbar
In [1]: a = 1
In [2]: x = (1,3)
In [3]: print(x)
(1, 3)
In [4]: who
a x
In [ ]:

“E' l'istess...“ *3/3*


stegua@DESKTOP-MH2Q2VS: ~
stegua@DESKTOP-MH2Q2VS:~$ python3.5
Python 3.5.2 (default, Nov 17 2016, 17:05:23)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> a = 1
>>> x = (1, 3)
>>> print(x)
(1, 3)
>>> exit()
stegua@DESKTOP-MH2Q2VS:~$
Windows PowerShell
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.
PS C:\Users\gualandi> ipython
Python 3.5.3 |Anaconda custom (64-bit)| (default, Feb 22 2017, 21:28:42) [MSC
Type "copyright", "credits" or "license" for more information.
IPython 5.1.0 -- An enhanced Interactive Python.
? -> Introduction and overview of IPython's features.
%quickref -> Quick reference.
help -> Python's own help system.
object? -> Details about 'object', use 'object??' for extra details.
In [1]: a = 1
In [2]: x = (1,3)
In [3]: print(x)
(1, 3)
In [4]: who
a x
In [5]: exit()
PS C:\Users\gualandi>

Corsi online (self-paced)

- Coursera:

  http://www.coursera.org/learn/python

- Udacity:

  http://www.udacity.com/course/programming-foundations-with-python--ud036

- EDX:

  http://www.edx.org/course/learn-program-using-python-utarlingtonx-cse1309x

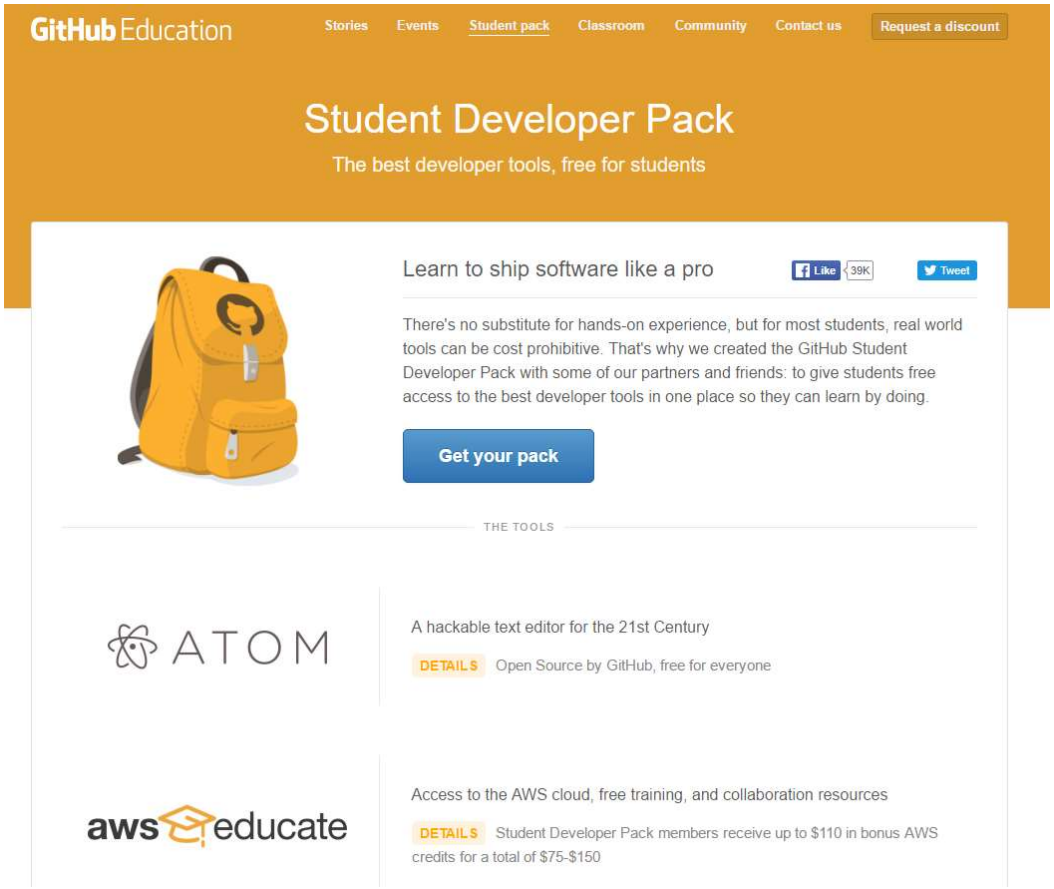
- Learn to code the hard way:

  http://learncodethehardway.org/python/

Altre risorse utili 1/2

GitHub Education Pack:

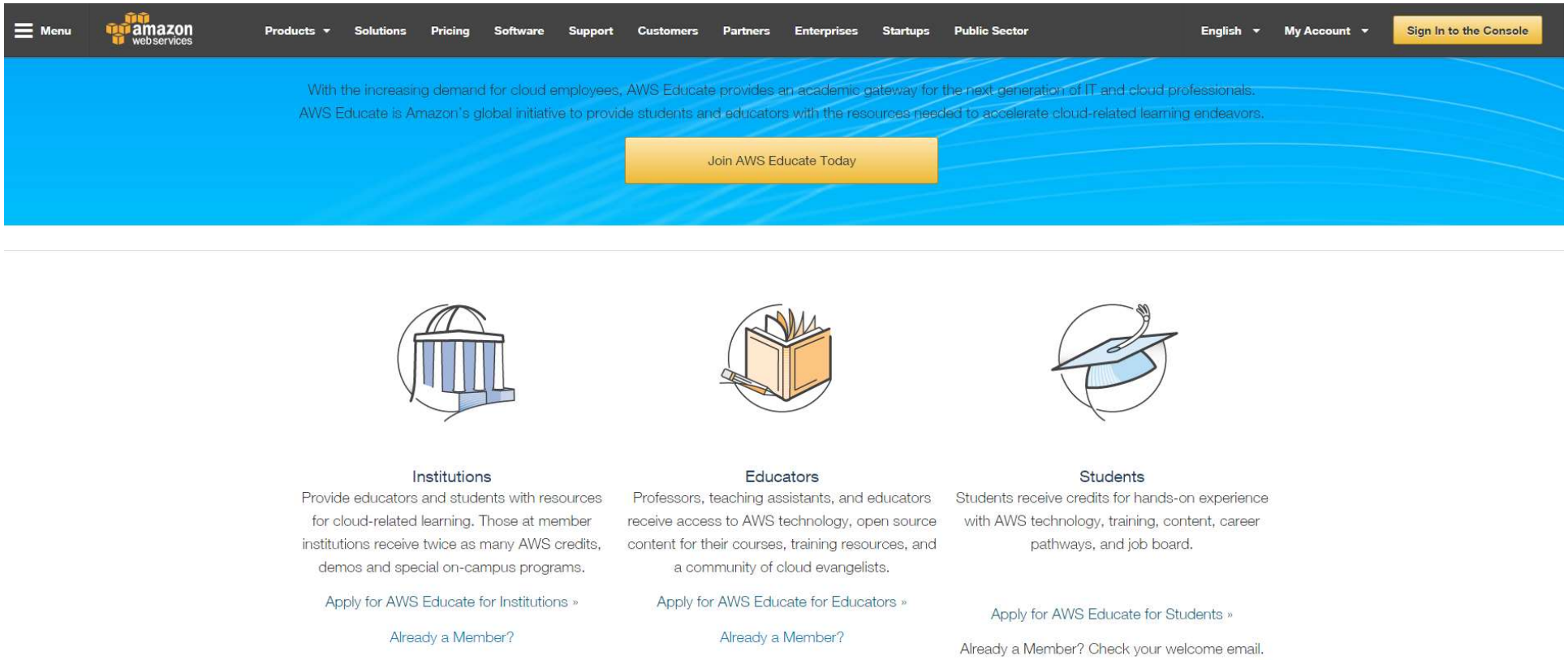
http://education.github.com/pack



Altre risorse utili 2/2

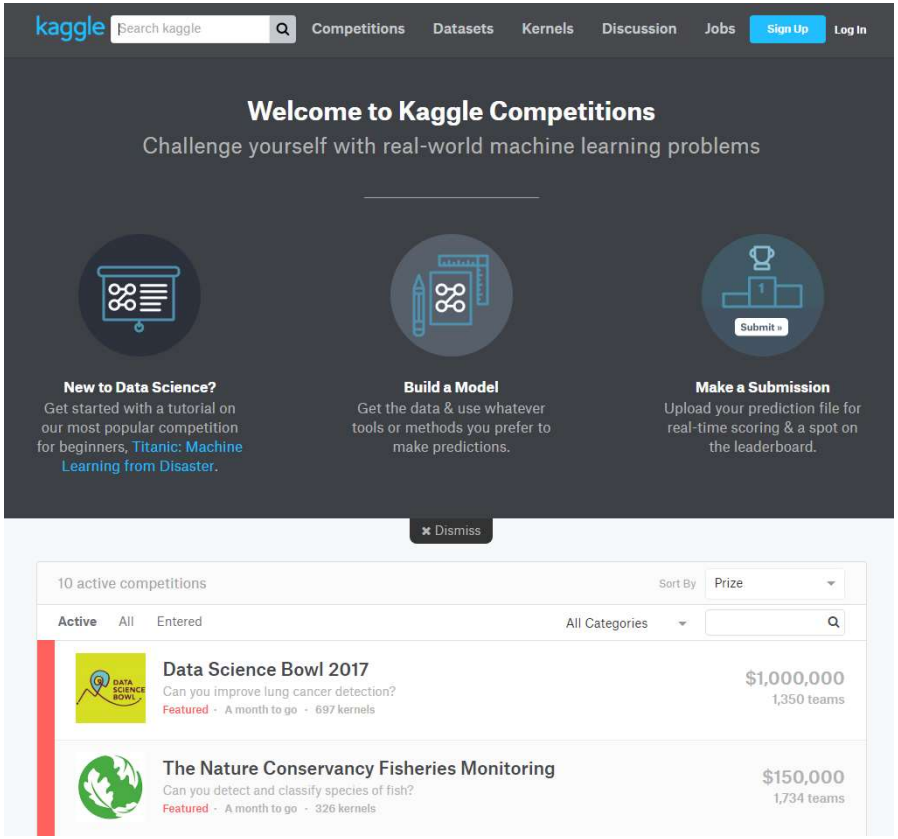
Amazon Web Services educate:

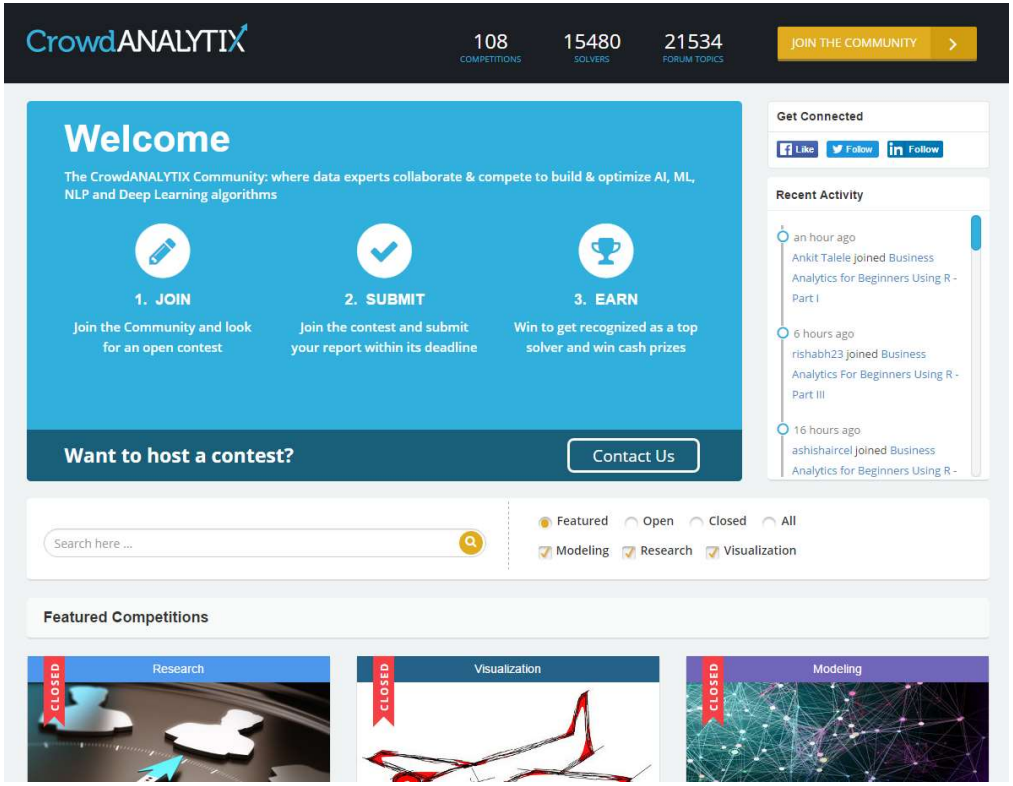
http://aws.amazon.com/education/awseducate/



Competizioni

- http://www.kaggle.com
- http://www.crowdanalytix.com





SymPy: Symbolic Mathematics