



POLITECNICO
MILANO 1863



SLIDES
DOPO IL
LABORATORIO

Fondamenti di Comunicazioni e Internet

Antonio Capone, Matteo Cesana, Guido
Maier, Francesco Musumeci



POLITECNICO
MILANO 1863



Lab 1: Scripting con Python

Soluzione Esercizio 1.2

**Antonio Capone, Matteo Cesana, Guido
Maier, Francesco Musumeci**

Ora tocca a voi!

Esercizio 1.2

Scrivere uno script che stampi il nome della pagina col **miglior tempo di risposta medio** tra **6** siti Internet. Per il calcolo del tempo medio, si definisca la funzione *media(list)* che ritorna la media dei valori contenuti in *list*.

- Numero di richieste = 10
- Siti internet:
 1. `http://www.google.com`
 2. `http://www.youtube.com`
 3. `http://www.polimi.it`
 4. `http://www.wikipedia.org`
 5. `http://www.amazon.com`
 6. `http://www.twitter.com`

Ora tocca a voi!

Esercizio 1.2

Hint:

LISTA.index(x) ritorna la posizione dell'elemento *x* nella lista *LIST*.

È l'inverso dell'accesso alla lista tramite posizione

```
L = [1, 5, 20, 4]          L = [1, 5, 20, 4]
print(L.index(5)) # STAMPA 1  print(L[1]) # STAMPA 5
```


NB: gli indici in Python iniziano da 0!

Soluzione Esercizio 1.2

```
1 import requests
2
3 siti = ['http://www.google.com', 'http://www.youtube.com', 'http://www.polimi.it',
4         'http://www.wikipedia.org', 'http://www.amazon.com', 'http://www.twitter.com']
5 avg = []
6 for url in siti:
7     print('Test', url)
8     tempi = []
9     for _ in range(10):
10        r = requests.get(url)
11        tempi.append(r.elapsed.microseconds/1000)
12    print('avg', sum(tempi)/len(tempi))
13    avg.append(sum(tempi)/len(tempi))
14
15 print(siti[avg.index(min(avg))], min(avg))
```

Soluzione Esercizio 1.2

```
1 import requests
2
3 siti = ['http://www.google.com', 'http://www.youtube.com', 'http://www.polimi.it',
4         'http://www.wikipedia.org', 'http://www.amazon.com', 'http://www.twitter.com']
5 avg = []
6 for url in siti:
7     print('Test', url)
8     tempi = []
9     for _ in range(10):
10        r = requests.get(url)
11        tempi.append(r.elapsed.microseconds/1000)
12    print('avg', sum(tempi)/len(tempi))
13    avg.append(sum(tempi)/len(tempi))
14
15 print(siti[avg.index(min(avg))], min(avg))
```



```
1 siti = ['SIT01', 'SIT02', 'SIT03']
2 avg = [150, 90, 200]
3 minimo = min(avg) # = 90
4 indice_minimo = avg.index(min(avg)) # = 1
5 indice_elemento_90 = avg.index(90) # = 1
6 sito_corrispondente = siti[avg.index(min(avg))] # = siti[1] = 'SIT02'
```