

## Parte 1

# Linguaggi dinamici: quando nascono?

## Nascita di un linguaggio

*L'invenzione dei linguaggi di programmazione riflette il periodo storico e le necessità dei programmatori*

- *Architettura dei calcolatori*
- *Applicazioni principali (killer-application)*
- *Esigenze di gestione / amministrazione*

*Come nascono i linguaggi dinamici (Perl, Python, Ruby, PhP, Javascript)?*

*Chi nasce prima e chi per ultimo?*

*Nascono dopo altri linguaggi?*

## Un po' di storia (1970-1980)

- Sistemi di grandi dimensioni (mainframe):
  - Applicazioni: calcolo scientifico
  - Interfacce: testuali
  - Amministrazione: scripting
- In questo contesto nascono:
  - C (veloce, adatto al calcolo, poco portabile)
  - Assembly (molto veloce, non portabile, adatto all'interfacciamento con l'hardware)
  - Shell (lento, portabile, adatto per manutenzione e configurazione dei sistemi)



## Un po' di storia (1980-1990)

- L'informatica prende piede su media e piccola scala
  - Singoli PC (Workstation)
  - Reti locali
- Sempre più persone hanno il compito di “amministrare” un calcolatore (*sistemisti*)
- Nasce il primo linguaggio di scripting vero e proprio: Perl (Larry Wall 1987)
  - Portabile e veloce
  - Orientato a manipolazione di file di testo (utile per gestire configurazioni e analizzare log)



## Un po' di storia (1990-2000)

- I programmatori cominciano a comprendere 2 grandi verità sull'architettura dei calcolatori
  - L'hardware delle macchine diventa più potente: ritmo simile alla legge di Moore (1965!)
  - Scrivere codice a basso livello diventa sempre più difficile, per via dell'aumentata complessità delle architetture
- Si diffonde il concetto di “linguaggio ad alto livello general purpose”
  - Python (Guido Van Rossum, 1991)
  - Ruby (Yukihiro Matsumoto, 1993)
  - Java (James Gosling, 1995)



## Un po' di storia (1990-2000)



- Invenzione (1991) e successivo boom (1995) del World Wide Web
- Nascono linguaggi nuovi per poter programmare applicazioni Web-based con maggiore semplicità rispetto al C
  - PHP (*PHP Hypertext Preprocessor*) nasce nel 1994 per creare pagine Web dinamiche
  - Javascript (Mocha/Livescript) nasce nel 1995 per essere usato in Netscape (primi browser Web)
- I linguaggi esistenti si “adattano”
  - C e Perl vengono tipicamente usati per i CGI (facilità accesso a DB e comunicazione di rete)

## Un po' di storia (2000-)

- L'uso dei linguaggi di alto livello si estende a progetti su larga scala
- Presa di coscienza dei programmatori: *il ciclo di sviluppo del software deve essere più semplice e breve*
- I linguaggi di alto livello sono corredati di una miriade di librerie per le funzioni più disparate
- Nasce il concetto di framework: linguaggio ad alto livello unito ad una serie di librerie general-purpose
  - .NET (compilatore C# - Visual Basic, Common Language Runtime e librerie)
  - Mono e Portable.NET (open source)

## Un rapido sguardo ai giorni nostri

**PYPL Index** - Popularity of Programming Language Index is created by analyzing how often language tutorials are searched on Google

Worldwide, Mar 2016 compared to a year ago:

Rank	Change	Language	Share	Trend
1		Java	24.2 %	+0.3 %
2	↑	Python	11.9 %	+1.2 %
3	↓	PHP	10.7 %	-0.8 %
4		C#	8.9 %	+0.1 %
5		C++	7.6 %	-0.5 %
6		C	7.5 %	+0.1 %
7		Javascript	7.3 %	+0.3 %
8		Objective-C	5.0 %	-0.9 %
9	↑↑	Swift	3.0 %	+0.4 %
10		R	2.9 %	+0.3 %
11	↓↓	Matlab	2.8 %	-0.3 %
12		Ruby	2.3 %	-0.2 %
13		Visual Basic	1.8 %	-0.4 %
14		VBA	1.5 %	+0.1 %
15		Perl	1.1 %	-0.1 %

Worldwide, Feb 2018 compared to a year ago:

Rank	Change	Language	Share	Trend
1		Java	22.55 %	-1.1 %
2		Python	21.3 %	+5.6 %
3		PHP	8.53 %	-1.8 %
4	↑	Javascript	8.49 %	+0.4 %
5	↓	C#	8.06 %	-0.6 %
6		C	6.51 %	-1.4 %
7	↑	R	4.23 %	+0.5 %
8	↓	Objective-C	3.86 %	-1.2 %
9		Swift	3.09 %	-0.4 %
10		Matlab	2.34 %	-0.5 %
11		Ruby	1.8 %	-0.4 %
12	↑↑↑	TypeScript	1.47 %	+0.5 %
13		VBA	1.46 %	+0.0 %
14	↓↓↓	Visual Basic	1.3 %	-0.3 %
15	↓	Scala	1.24 %	+0.1 %

# Linguaggi dinamici: definizione e caratteristiche

## Definizioni

**Possibile definizione (da Wikipedia)**

- **Linguaggio statico: è un linguaggio ad alto livello in cui le operazioni effettuate a tempo di esecuzione (run time) sono legate (*quasi*) esclusivamente all'esecuzione di codice (*se si esclude la gestione dello stack*)**
  - **C, assembly**

# Definizioni

Possibile definizione (da Wikipedia)

- **Linguaggio statico:** è un linguaggio ad alto livello in cui le operazioni effettuate a tempo di esecuzione (run time) sono legate (*quasi*) esclusivamente all'esecuzione di codice (*se si esclude la gestione dello stack*)
  - C, assembly
- **Linguaggio dinamico:** è un linguaggio ad alto livello in cui le operazioni effettuate a tempo di esecuzione (run time) non sono legate esclusivamente all'esecuzione di codice
  - Perl, Python, Ruby, PHP, Javascript

Java?

## Linguaggi dinamici - definizione

- Non esiste in realtà una definizione univoca, ma un insieme di caratteristiche tipiche
  - *Tipizzazione dei dati*
  - *Metaprogramming*
  - *Gestione dinamica di memoria ed errori*
  - *Modello di generazione del codice (prodotto intermedio es. bytecode)*
- Alcune caratteristiche si possono ritrovare in altri linguaggi non considerati propriamente dinamici
  - Es. Java

Operazioni  
svolte a run-time

## Linguaggio statico

- Ha una fase di compilazione, in cui il codice sorgente viene tradotto in un formato a basso livello esclusivo per l'architettura considerata
  - La traduzione da codice sorgente a codice macchina è 1:1 (rappresentazione fedele)
- Permette l'esecuzione ad una velocità elevata
- I tipi di dati sono identificati a tempo di compilazione e non sono mutabili a tempo di esecuzione
- Non fornisce strumenti di controllo, né semplificazioni, quasi tutto è lasciato al programmatore (memoria, tipizzazione)

## Linguaggio dinamico

- Ha una fase di compilazione, in cui il codice sorgente viene tradotto in un formato intermedio indipendente dall'architettura (es. bytecode)
  - Il formato intermedio è interpretato → linguaggio portabile
- L'interprete si serve di funzioni interne per gestire memoria ed errori in modo automatico a run-time
  - Assenza di dettagli ostici per il programmatore
- Ha una tipizzazione dinamica dei dati
  - I tipi di dato possono mutare a run-time

# Linguaggio dinamico

- Ha la caratteristica di sapersi “analizzare” e “modificare” durante l'esecuzione (Metaprogramming)
  - Eseguire funzioni diverse a seconda delle condizioni operative a run-time
  - Cambiare il codice stesso del programma
  - Creare strutture dati variabili nel tempo – spesso eterogenee
- Presenza massiccia di librerie esterne facilmente utilizzabili per diversi compiti
  - Servizi di calcolo scientifico, interfacce grafiche complesse, supporto per il Web

## Linguaggi dinamici vs statici

- **Pro:**
  - La scrittura di un software diventa la scrittura del suo scheletro (presenza delle librerie esterne → forte riutilizzo del codice)
  - La scrittura di codice è estremamente semplificata rispetto ai linguaggi statici
  - Flessibile e portabile
  - Adatti per la prototipazione veloce
- **Contro:**
  - Più lento di un linguaggio statico
    - Molte operazioni a run-time
  - Può incoraggiare la programmazione “sloppy”

Sloppy...





# Perchè focalizzarci su Python?

Worldwide, Feb 2018 compared to a year ago:

Rank	Change	Language	Share	Trend
1		Java	22.55 %	-1.1 %
2		Python	21.3 %	+5.6 %
3		PHP	8.53 %	-1.8 %
4	↑	Javascript	8.49 %	+0.4 %
5	↓	C#	8.06 %	-0.6 %

Python è il linguaggio dinamico più usato attualmente

## Perchè Python?

### • Usato nei più popolari siti Web (server-side)

Programming languages used in most popular websites\*

Websites	Popularity (unique visitors per month) <sup>[1]</sup>	Front- end (Client- side)	Back-end (Server- side)	Database	Notes
Google.com <sup>[2]</sup>	1,600,000,000	JavaScript	C, C++, Go, <sup>[3]</sup> Java, Python	Bigtable, <sup>[4]</sup> MariaDB <sup>[5]</sup>	The most used search engine in the world
Facebook.com	1,100,000,000	JavaScript	Hack, PHP (HHVM), Python, C++, Java, Erlang, D, <sup>[6]</sup> Xhp, <sup>[7]</sup> Haskell <sup>[8]</sup>	MariaDB, MySQL, <sup>[9]</sup> HBase Cassandra <sup>[10]</sup>	The most visited social networking site
YouTube.com	1,100,000,000	JavaScript	C, C++, Python, Java, <sup>[11]</sup> Go <sup>[12]</sup>	Vitess, BigTable, MariaDB <sup>[5]</sup> <sup>[13]</sup>	The most visited video sharing site

GO

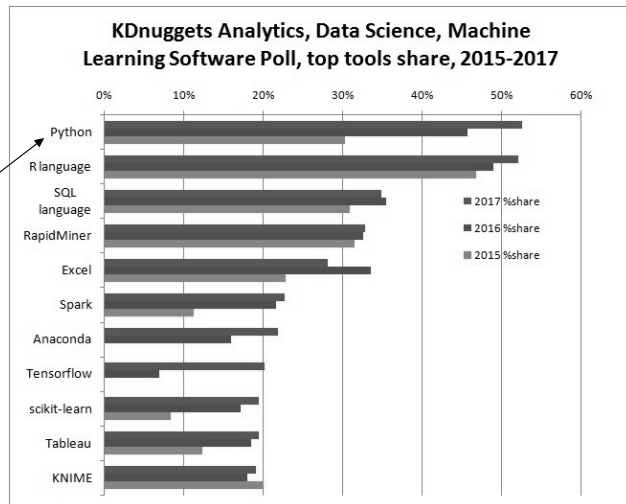
\*dati 2018

# Perchè Python?

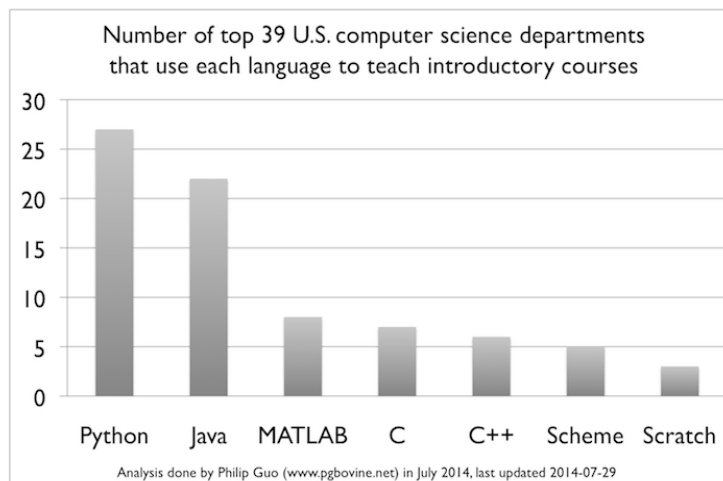
- Popolarità in ascesa
- E' il linguaggio dinamico di riferimento per la piattaforma dei servizi offerti da Google
- Curva di apprendimento ripida
- Flessibilità
- Semplicità

Popolare nel contesto della **analisi dati**  
Interfacciabile con software scientifico R

Linguaggi dinamici



## Trend delle Università Americane



**Python ha superato Java (linguaggio predominante per gli ultimi 10 anni)**

Linguaggi dinamici

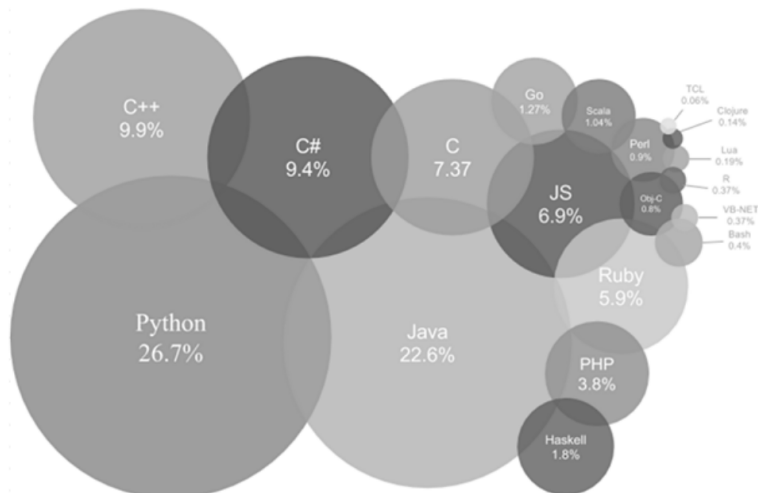
Claudia Canali

20

# Rilevazioni CodeEval

**CodeEval = comunità di sviluppatori interessati a risolvere sfide di programmazione (contest)**

Most Popular Coding Languages of 2016



**Nota:** le sfide possono essere risolte in un linguaggio a piacere

21

## Esempi

```
# python_ex.py
print("Hello World!\n") #py3
print "Hello World!\n" #py2.7
var1 = 42
var1 = var1 + 1
print(var1) #py3
print var1 #py2.7
```

```
# python python_ex.py
```

```
# ruby_ex.rb
puts "Hello World!\n"
var1 = 42
var1 = var1 + 1
puts var1
```

```
# ruby ruby_ex.rb
```

```
# perl_ex.pl
print "Hello World!\n";
$var1 = 42;
$var1++;
print "$var1\n";
```

```
# perl python_ex.pl
```

```
# php_ex.php
<?php
print "Hello, World!\n";
$x = 42;
$x = $x + 1;
print $x;
?>
```

```
# php php_ex.php
```

# Esempi

```
# javascript_ex.html  
<!DOCTYPE HTML>  
<html>  
<body>  
  <p>Header...</p>  
  <script>  
    alert('Hello, World!')  
    var a = 42;  
    a = a + 1;  
    alert(a)  
  </script>  
  <p>...Footer</p>  
</body>  
</html>
```

```
# firefox javascript_ex.html
```