



Computación

Parte II:

Aprendiendo a programar

con Python

```
class BigFile:
    def __init__(self, datadir, ndims):
        idfile = os.path.join(datadir, "id.txt")
        self.names = [x.strip() for x in str.split(open(idfile).read()) if x.strip()]
        self.name2index = dict(zip(self.names, range(len(self.names))))
        self.ndims = ndims
        self.featurefile = os.path.join(datadir, "feature.bin")
        print "[BigFile] %d features, %d dimensions" % (len(self.names), self.ndims)
        print "      binary: %s" % self.featurefile
        print "      txt: %s" % idfile

    def read(self, requested, isname=True):
        if isname:
            index_name_array = [(self.name2index[x], x) for x in requested if x in self.names]
        else:
            assert(min(requested) >= 0)
            assert(max(requested) < len(self.names))
            index_name_array = [(x, self.names[x]) for x in requested]
            index_name_array.sort()

        vecs = seq_read(self.featurefile, self.ndims, [x[0] for x in index_name_array])
        return [x[1] for x in index_name_array], vecs

    def shape(self):
        return [len(self.names), self.ndims]
```



Agenda

1. ¿Qué es Python?
2. Otros lenguajes de programación
3. ¿Qué es un programador?
4. Tipos de programadores
5. Características de Python
6. Instalación
7. IDEs y alternativas en la nube
8. ¿Porqué tengo que aprender a programar?



¿Qué es Python?

- Python es un lenguaje de programación ampliamente utilizado en las aplicaciones web, el desarrollo de software, la ciencia de datos y el machine learning (ML).
- Los desarrolladores utilizan Python porque es eficiente y fácil de aprender, además de que se puede ejecutar en muchas plataformas diferentes



¿Qué es Python?

- Python es de código abierto, en constante cambio y mantenimiento por millones de desarrolladores a lo largo del mundo
- Ampliamente usado, algunos ejemplos



Otros lenguajes de programación

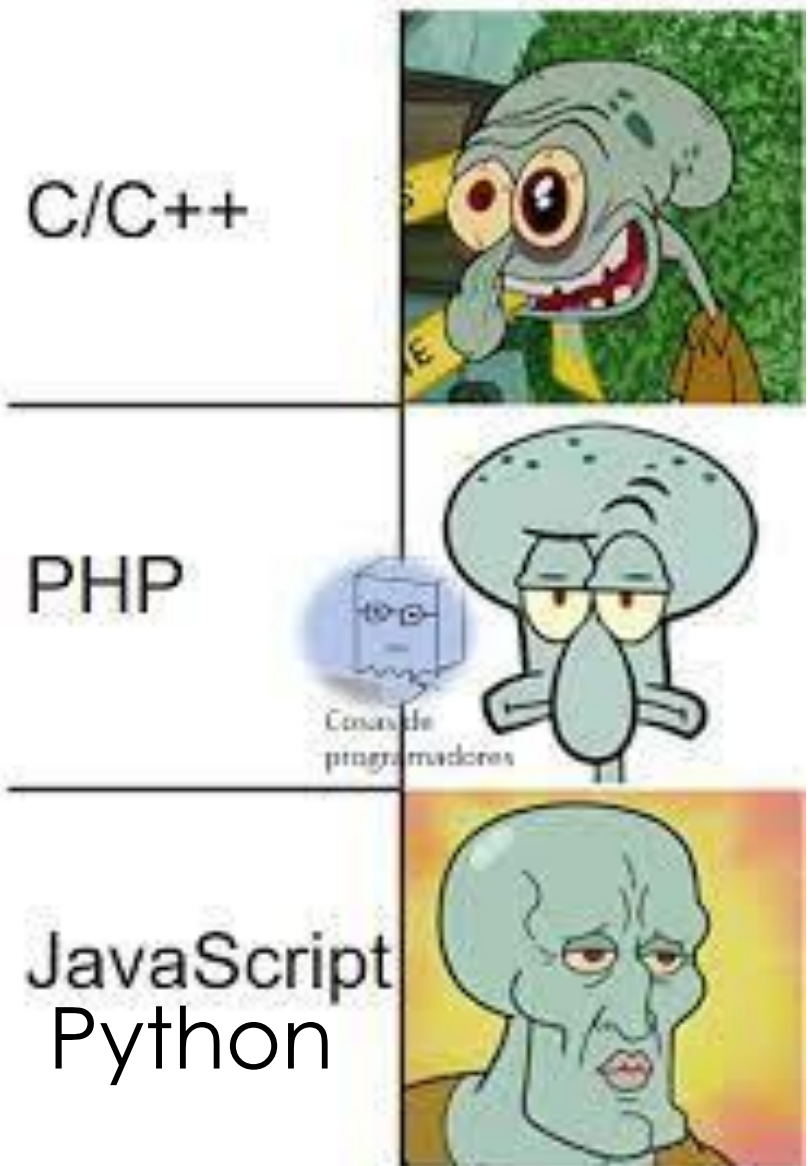


- Muy probablemente se encuentren con colegas que aprendieron otro lenguaje. En física Fortran era el santo grial para desarrollar métodos numéricos, ahora Julia y Python están sucediéndolo.
- Lo importante no es programar en 5000 lenguajes, si no más bien “pensar como un programador” y conocer los tipos de estructuras. Una vez dominado esto, el aprender otro lenguaje es muchísimo más sencillo.



¿Qué es un programador?

Un programador es un profesional que se encarga de escribir, de depurar y de revisar todo el código fuente de un software informático para que lleve a cabo determinadas tareas, o incluso para que mejore, incorporando nuevas funciones para hacer que sea una herramienta más completa y eficiente.



Tipos de programadores

Existen al menos 6 tipos de programador

1. Desarrollador Frontend:

Encargado de la aplicación que interactúa con el usuario (interfaz y experiencia de usuario)

2. Desarrollador Backend

Encargado del diseño, implementación, lógica funcional, escalabilidad y seguridad de la app

3. Desarrollador Fullstack

Front + Back

4. Desarrollador Desktop

Desarrolla aplicaciones que se ejecutan en ordenadores o laptops (como el office)

5. Desarrollador móvil

Desarrolla aplicaciones que se ejecutan en los teléfonos, relojes inteligentes y tablets

6. Data Scientist

Crean herramientas para manejar, utilizar y analizar grandes volúmenes de información, con el fin de obtener conclusiones e información valiosa

Características de Python

- Es un lenguaje interpretado

El código se ejecuta línea por línea de arriba hacia abajo*. En caso de existir un error la ejecución del código se detendrá, de esta forma es fácil encontrar errores.

- Es un lenguaje fácil de utilizar y aprender

Python usa palabras similares al inglés, no es necesario colocar llaves o puntos y comas, pero sí es sensible a los espacios de indentación

- Es un lenguaje tipeado dinámicamente

No es necesario declarar que tipo de variable se usa, Python puede inferirlos al momento de ejecutar el código

- Es un lenguaje de alto nivel

Python es de alguna manera más similar a un idioma humano comparado contra otros lenguajes, entonces uno no se debe preocupar por asuntos de memoria o arquitecturas “escondidas”

- Es un lenguaje multiparadigma

Las piezas de código pueden considerarse como objetos (objeto coche, balón, etc) esto es el paradigma orientado a objetos. Pero pueden también considerarse como funciones que es el paradigma funcional. Admite diversos paradigmas que dado un problema son más sencillos de resolver en ciertos paradigmas. No entraremos más en detalle con esto pues son temas más avanzados.

OK chido pero ¿Dónde se programa o qué?

- Primero es necesario instalar Python. En caso de usarse Linux, Python ya está instalado.
- Se recomienda instalar anaconda pues trae consigo varias herramientas útiles además de Python. Anaconda no viene instalado por defecto en Linux.

[Products ▾](#)[Pricing](#)[Solutions ▾](#)[Resources ▾](#)[Partners ▾](#)[Blog](#)[Company ▾](#)[Contact Sales](#)

Data science technology for a better world.

Anaconda offers the easiest way to perform Python/R data science and machine learning on a single machine. Start working with thousands of open-source packages and libraries today.

[Download !\[\]\(95b425611cbd2b8716a140cf67c81822_img.jpg\)](#)

For Windows

Python 3.9 • 64-Bit Graphical Installer • 621 MB

Have you registered for c
webinar? 🤖

IDES



- Escribir código puede hacerse en cualquier editor de texto, como bloc de notas, vim, nano, etc, Sin embargo, hacerlo en éstos suele ser tedioso y no ofrece funcionalidades como ejecutar código, entre otros.
- Por esta razón todos los programadores de cualquier lenguaje usan uno o varios IDES para desarrollar código.
- Un IDE (entorno de desarrollo integrado) es una aplicación que ofrece varias funcionalidades para facilitar la vida de un programador. Normalmente consiste de un editor de código fuente, compilador y depuador
- Existe una gran variedad de IDES como NETBean, Eclipse, Visual Studio Code, Notepad ++ entre otros.
- Nosotros usaremos Jupyter Notebook como IDE pues es una alternativa intuitiva y directa de usar con Python
- También veremos un poco de Visual Studio Code, ya que es uno de los IDE más popular que contiene una gran cantidad de herramientas de desarrollo

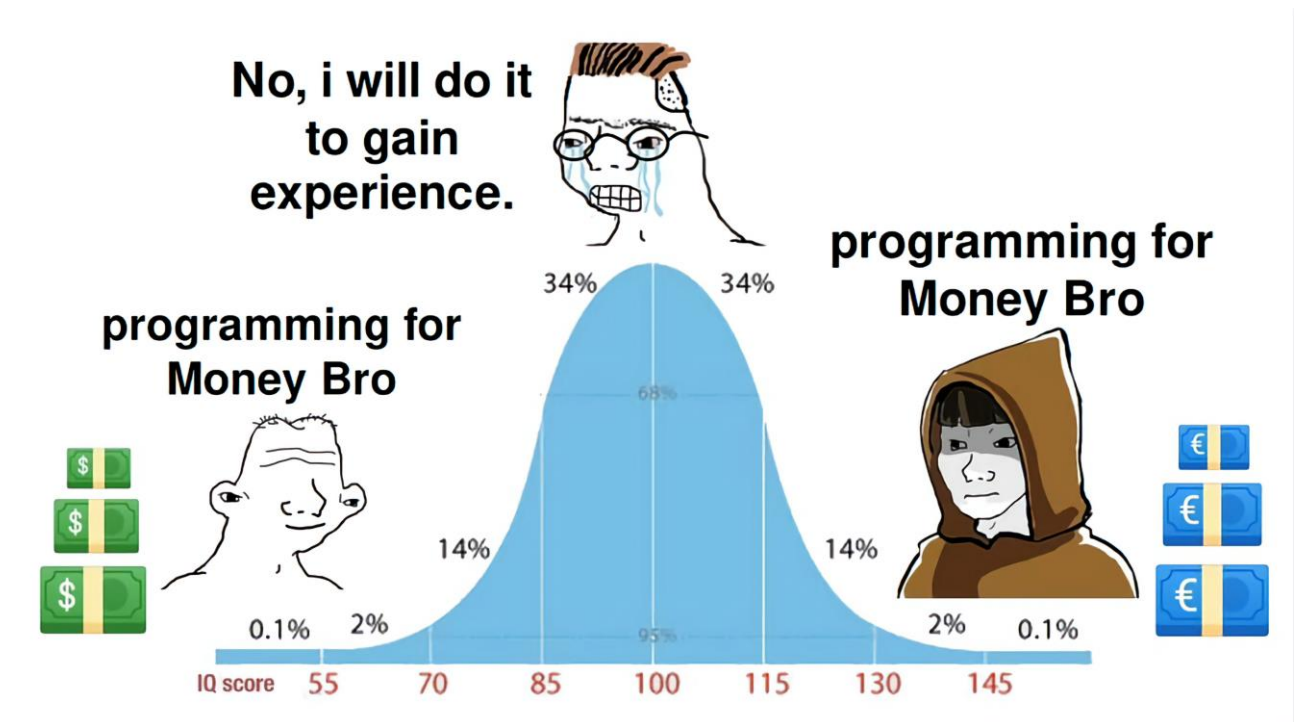
No quiero instalar nada en mi equipo

- Una alternativa que permite trabajar con Python sin la necesidad de instalar software es Google Colab, el cual provee de una instancia en jupyter notebook para comenzar a trabajar



¿Porqué tengo que aprender a programar?

- Esta pregunta ya se las respondí las primeras clases
- Programar ya no es un lujo es una necesidad para la ciencias e ingenierías
- Prácticamente cualquier tarea que sea repetitiva puede programarse y ahorrarnos mucho sufrimiento.
- La computadora ofrece resultados, nosotros debemos saber interpretarlos, esto es, por ahora, una herramienta muy poderosa.
- Y “la más importante” -->



A large, irregular pink brushstroke shape with a rough, textured edge, resembling a paint splatter or a hand-drawn shape. It is positioned on the left side of the slide.

*Ahora si,
manos al barro!*

A black square graphic with white text. The text 'Let's' is at the top, 'Code' is in the middle and larger, and 'Together' is at the bottom. The text is enclosed in large white curly braces on either side.

Let's
Code
Together