**¿Por qué Python?**

El motivo principal por el que cada vez más desarrolladores evolucionan hacia Python es la emergencia de soluciones tales como **Django** (framework para aplicaciones web gratuito y open source, escrito en Python), **Twisted (**es un framework de red para programación dirigida por eventos escrito en Python), o el hecho de que Python se haya impuesto como la referencia indiscutible en otros dominios, como por ejemplo la construcción de **interfaces gráficas** para aplicaciones Gnome, Kde o Windows (entornos de escritorio). También es la única alternativa libre y potente.

Python abarca sobre prácticamente todos los segmentos: desde la programación de sistema hasta la programación web, pasando por la programación de aplicaciones sin interfaz gráfica, los videojuegos, las redes, el cálculo científico, el Big Data, etc.

Además otro motivo para aprender Python es conocer el entramado de programación del ERP Odoo, es cual está programado en Python. Dicho ERP es libre y muy potente.

**Python es multiparadigma**

Un paradigma es un modelo de programación. En el sentido informático, es el conjunto de reglas gramaticales y herramientas que permiten a un desarrollador describir algoritmos, .

Python es multiparadigma**: imperativo** (COBOL, BASIC, y C), **orientado a objetos** (C++, Java, JavaScript), **orientado a componentes** (crear bloques reutilizables, Patrones de diseño), **orientado a eventos** (principalmente para interfaces gráficas en aplicaciones Web), **orientado a aspectos** (permite desacoplar el aspecto), **paradigma funciona**l (se centra en los datos), **paradigma lógico, paradigma concurrente, programación por contrato**.

Los programas escritos en Python se ejecutan en una **máquina virtual** y estas máquinas existen en muchas arquitecturas y hace que cualquier programa en Python sea extremadamente portable. Además, para la distribución de su programa, existen herramientas para empaquetarlo (crear un instalador para Windows o un paquete para Linux, por ejemplo).

Python es un lenguaje **dinámicamente tipado**, permite una mayor flexibilidad (modificar el tipo de una variable en tiempo de ejecución, por ejemplo) y es una referencia absoluta para los lenguajes dinámicamente tipados gracias a sus soluciones innovadoras para garantizar cierta seguridad de programación.

Es **libre** y **gratuito**, funciona sobre todas las plataformas y permite realizar scripts de administración del sistema.

## Fases de ejecución de un programa Python

Cuando se inicia un programa Python, **la máquina virtual Python se arranca**. Realiza la interfaz entre el programa Python y el sistema operativo. Posteriormente se **compila (**también los módulos que importa)y se **ejecuta**.

Para evitar compilar de nuevo los módulos con cada uso del script, su versión compilada se escribe en un archivo **.pyc** y, con cada nueva ejecución del script, se verifica si los módulos no se han modificado, en cuyo caso se realiza una nueva compilación. Por el contrario, el propio módulo principal se compila cada vez, sistemáticamente, al vuelo, y no se guarda en ningún archivo.