

# PRÁCTICA 1

INTRODUCCIÓN A LA PROGRAMACIÓN

**Nombre:** Hugo Pelayo

**Asignatura:** Programación

**Curso:** 2022-2023

# Índice

Investiga sobre los lenguajes Python y Java .....	3
Historia y evolución de Python.....	3
Características de Python .....	3
Qué se necesita para comenzar a programar en Python .....	4
Configuración del entorno Visual Studio Code con Python .....	5
Historia y evolución de Java .....	6
Características de Java .....	7
Qué se necesita para comenzar a programar.....	8
Configuración del entorno Visual Studio Code con Java .....	8

## Investiga sobre los lenguajes Python y Java

### Historia y evolución de Python

Python es un lenguaje de programación de alto nivel interpretado cuya filosofía se enfoca en la legibilidad de código, es usado para el desarrollo de todo tipo de aplicaciones como Netflix, Spotify, entre otros. Es un lenguaje que soporta varios paradigmas de programación, a destacar, soporta parcialmente la programación orientada a objetos, la programación imperativa y con poco soporte, la programación funcional.

Fue creado por Guido Van Rossum en el *Centrum Wiskunde and Informatica*, en los Países Bajos. Python surgió como sucesor de ABC, un lenguaje capaz de controlar excepciones e interactuar con Amoeba, un sistema operativo basado en una arquitectura de micronúcleo. En 20 de febrero de 1991, van Rossum publicó la primera versión de Python en alt.sources. Ya constaba ofrecía soporte para herencia sobre clases, para el manejo de excepciones, para funciones e incorporaba los tipos modulares (clases) "str" (para el tratamiento de cadenas de caracteres, "list" (para el manejo de arreglos de elementos), "dict" (de diccionario, estructura para el almacenamiento de elementos por clave y valor), entre otros. En el año 1994 se formó el principal foro de discusión de Python, cosa que apoyo al crecimiento de usuarios de este lenguaje. En ese año también se publicó la versión 1.0 de Python que ofrecía las herramientas principales para el soporte de la programación funcional en Python, a destacar, los elementos "lambdas" (funciones sin nombre), "reduce", "filter", y "map".

Python 2.0 surgió en el año 2000, lanzado por BenOpen.com. Esta versión de Python también introduce un elemento característico del lenguaje de programación funcional Haskell, las listas de comprensión. Aunque la sintaxis es muy similar, Python las implementa de una manera diferente. Otro aspecto importante a destacar es que esta versión introduce un sistema de recolección de basura (*garbage collector* en inglés), un sistema que ayuda a la gestión de memoria. Python 2.2 unifica los tipos de Python, también los generadores, una rutina especial que se puede usar para controlar el comportamiento de un iterador en un bucle, característica heredada del lenguaje Icon. Python 2 fue oficialmente discontinuado con la versión 2.7.x el 1 de enero de 2020, lo que implica que las versiones de la misma rama dejarían de recibir soporte desde entonces. En la actualidad solo reciben actualizaciones la rama de Python 3.6.x y versiones posteriores a esta.

### Características de Python

Python es un lenguaje de programación multiparadigma, esto es, que permite varios estilos de programación sin forzar al programador a optar por uno en concreto.

Utiliza el tipado dinámico, es decir, sus variables pueden tomar valores de distintos tipos durante la ejecución del programa. Utiliza un sistema de conteo de referencias (en inglés *reference counting*) para la gestión de memoria, entonces los objetos que ocupan memoria se liberan

cuando el número de referencias a este se reduce a 0, ninguna variable contiene el objeto en este caso.

## Qué se necesita para comenzar a programar en Python

Para empezar a programar en Python primero necesitamos crearnos un entorno de programación en nuestro sistema operativo de preferencia. Para ello necesitaremos primero instalar el intérprete de Python, es recomendable descargar siempre la última versión estable que podremos encontrar siempre en su página oficial. Los pasos a seguir serían:

1. Abrimos nuestro navegador favorito y accedemos a Google.
2. Buscamos por Python, seleccionamos la opción "Download Python - Python.org" y nos dirigirá a la página oficial de descarga de Python.
3. En Windows nos saldría el mensaje "Download the latest version for Windows", hacemos clic sobre el botón amarillo justo debajo de este mensaje (Download Python 3.10.x) y se

empezará a descargar Python.

4. Nos dirigimos a la carpeta donde se ha realizado la descarga y abrimos el ejecutable para comenzar el proceso de instalación.
5. Ya en la ventana de instalación dejamos marcadas las dos opciones "Install launcher for all users (recommended)" y "Add Python 3.10.x to PATH", estas opciones son para no reinstalar Python por cada nuevo usuario del sistema y para que poder invocar el intérprete de Python desde la línea de comandos, aun así esta última no suele ser recomendable sobre cuando todo cuando tenemos varias versiones del intérprete de Python las cuales nos gustaría poder usar de manera simultánea desde la línea de comandos.
6. Finalmente hacemos clic sobre "Install Now" y esperamos a que acabe el proceso de instalación.
7. Una vez instalado, abrimos nuestro cmd (Tecla Windows y luego escribimos "cmd", nos saldrá el Símbolo de Sistema). Ahora escribimos "Python --version" y nos debería mostrar una serie de números en pantalla, que sería la versión de nuestro intérprete, por ejemplo, 3.10.7.

Una vez instalado el intérprete, lo podemos aprovechar en entornos de desarrollo interactivos como *PyCharm*, *Visual Studio Community* o *Visual Studio Code* para poder ejecutar, editar y depurar nuestros códigos con más comodidad.

Python ofrece varias guías para programadores, entre ellas una guía para principiantes: [Beginners Guide](#).

Para crear un simple programa en Python, simplemente abrimos nuestro bloc de notas y escribimos una secuencia de sentencias, podemos escribir `print("Hello World!")` para simplificar. Ahora guardamos el archivo con extensión `.py`. Seguido de este paso, abrimos una terminal, en el directorio donde esta nuestro archivo y ejecutamos el comando `Python nuestro_script.py`.

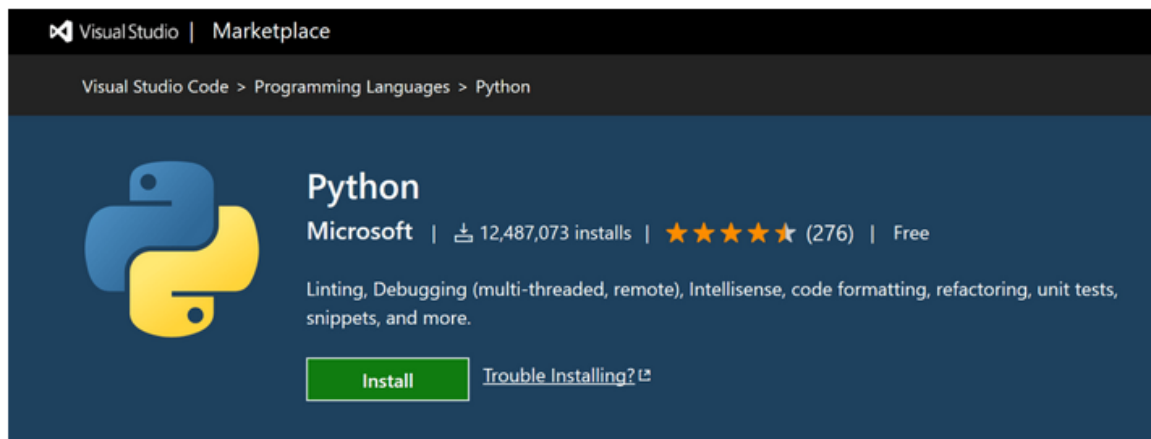
## Configuración del entorno Visual Studio Code con Python

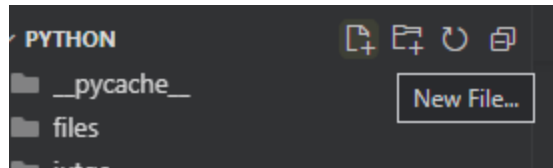
Primero de todo, nos instalaremos un intérprete realizando los pasos indicados en el anterior apartado. Después de esto procedemos a descargar VS Code a través de su [página web oficial](#).

Deslizamos hacia abajo hasta la sección "Install Visual Studio Code and the Python Extension". Descargamos a instalamos VS Code, la instalación es sencilla a intuitiva y nos instalamos la extensión de Python.

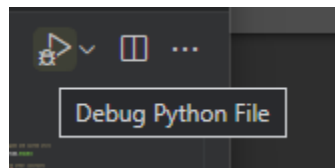
## Install Visual Studio Code and the Python Extension

1. If you have not already done so, install [VS Code](#).
2. Next, install the **Python extension for VS Code** from the Visual Studio Marketplace. For additional details on installing extensions, see [Extension Marketplace](#). The Python extension is named Python and it's published by Microsoft.

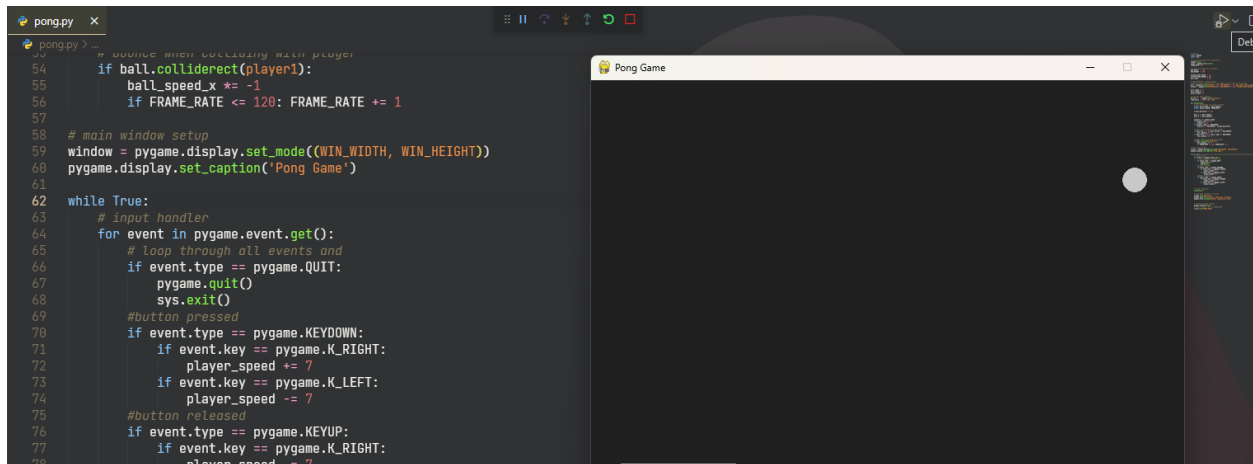




Ahora podemos abrir VS Code y crear un nuevo script. Para ejecutarlo hacemos click sobre el botón de reproducir de la esquina superior derecha de nuestra ventana de VS Code.



Ahora podremos ver nuestro programa en ejecución. En la imagen de abajo se muestra un pequeño juego de (Pong) creado por el autor de este artículo usando el módulo Pygame de Python.



## Historia y evolución de Java

Es un lenguaje de programación muy utilizado en el desarrollo de aplicaciones web, es multiplataforma y soporta principalmente el paradigma de la programación orientada objetos. Se utiliza también para el desarrollo de aplicaciones móviles y de escritorio. Java en sí es considerado una plataforma por ser un sistema que sirve como base para hacer funcionar ciertos módulos de hardware y software con los que es compatible. Fue comercializado por primera vez en 1995 por Sun Microsystems y desarrollado originalmente por James Gosling. Su sintaxis deriva en gran parte de C y C++, pero ofrece menos soporte para utilidades de bajo nivel que ambos. Las aplicaciones en Java se compilan a bytecode que puede interpretar cualquier máquina virtual de Java indistintamente de la arquitectura del computador en cuestión.

Sun definió por primera vez la implementación de referencia original para los compiladores de Java, máquinas virtuales y librerías, trabajo que se publicó en 1995.

Originalmente Java se creó para utilizar en un proyecto sobre un decodificador de señales de televisión en la operación The Green Project en el año 1991. En un principio se llamó Oak, se pasó a llamar Green tras darse cuenta de que Oak ya existía y pertenecía a una marca comercial de adaptadores de tarjetas gráficas, entonces se acabó llamando Java. Hay muchas teorías sobre la elección de este último nombre, sin embargo, destaca más el hecho de que el nombre tiene origen en un tipo de café de una cafetería que frecuentaban sus desarrolladores. El objetivo en un principio era diseñar un lenguaje de programación con una estructura similar a la de C++, aún así en 1994, sus desarrolladores acabaron orientándolo al entorno Web, porque creyeron que el navegador web Mosaic haría del internet un medio más interactivo, concepto que se asimilaba mucho a la televisión por cable, idea inicial del proyecto de Java. Naughton, uno de los desarrolladores de Java, creó entonces un prototipo de navegador, WebRunner, que más tarde se conocería con el nombre de HotJava.

La promesa inicial de Gosling era "Write once, Run Anywhere" (Escríbelo una vez, ejecútalo en cualquier plataforma), un concepto que promovía la transportabilidad, es decir, la capacidad de ejecutar los mismos códigos en cualquier plataforma que ofreciese soporte para el entorno de ejecución de Java, pudiendo compartir de esta forma los mismos binarios si tener que reescribir el código fuente.

Desde la versión 1.0 del JDK (*Java Development Kit*), lanzada en enero de 1996, Java ha experimentado números cambios sobre todo en las clases que ofrece su librería estándar. En diciembre de 1998 se lanzó el JDK 1.2 bajo los nombres de *Java 2* y *J2SE (Java 2 Platform, Standard Edition)*, esto con el objetivo de distinguir la plataforma base de las ediciones *J2EE (Java 2 Platform, Enterprise Edition)* y *J2ME (Java 2 Platform, Micro Edition)*.

## Características de Java

Entre las características de Java podemos destacar que es fuertemente orientado a objetos, en efecto, el programa principal es un objeto que viene encapsulado en forma de clase. Aquí los objetos son generalmente clases que constan de atributos y los métodos que nos sirven para poder operar sobre estos atributos. Otro aspecto importante a destacar en este aspecto es el soporte para objetos genéricos que se pueden reaprovechar para otras aplicaciones, cosa que agiliza el desarrollo de estas.

Java ofrece independencia de plataforma, por tanto, programas escritos en este lenguaje se pueden ejecutar en cualquier plataforma indistintamente del hardware, siempre y cuando haya soporte para la máquina virtual de Java en la plataforma destino. Para ello los compiladores de Java, traducen los archivos de código fuente de Java a binarios con extensión *.class*, los cuáles contienen *bytecode*, un tipo de instrucciones interpretables por la máquina virtual de Java.

A diferencia de C++, lenguaje del cuál Java hereda muchos aspectos sobre el paradigma de la programación orientada a objetos, Java proporciona un recolector de basura para solucionar el

problema de las fugas de memoria, error muy común entre programadores de C++. Las fugas de memoria suceden cuando alojamos memoria de forma dinámica (en tiempo de ejecución) y no la desalojamos acabado su uso, esto se convierte entonces en memoria ocupada pero no aprovechable. El *garbage collector* (recolector de basura) de Java se ocupa de este problema.

## Qué se necesita para comenzar a programar

Para empezar a desarrollar aplicaciones en Java lo primero que debemos hacer es decidarnos por una edición, la que más se ajuste a nuestras necesidades (mencionadas con anterioridad). Tenemos entonces a elegir: J2ME, J2SE o J2EE. Esta última se reserva para aplicaciones web más complejas con accesos a bases de datos, entre otros aspectos. Para aplicaciones menos complejas y más o menos completas que se puedan ejecutar en PC, la edición J2SE sería suficiente.

Ya escogido J2SE, debemos descargarnos el SDK (*Software Development Kit*) de Java que incluye herramientas para desarrollo de aplicaciones Java como el compilador, el depurador, entre otras.

## Configuración del entorno Visual Studio Code con Java

Para configurar el entorno de Visual Studio Code en nuestro ordenador debemos instalar el editor de texto en cuestión, VS Code, seguido del JDK (*Java Development Kit*) y las extensiones de Java necesarias. Nos dirigimos entonces a la página del siguiente enlace: [VS Code Tutorial](#).

The screenshot shows the Visual Studio Code website. The top navigation bar includes links for Visual Studio Code, Docs, Updates, Blog, API, Extensions, FAQ, and Learn. A search bar and a 'Download' button are also present. The main content area is titled 'Getting Started with Java in VS Code' and includes a sidebar with navigation links. The main text describes the tutorial's purpose and provides links to the 'Coding Pack for Java' and 'Installing extensions'. The right sidebar contains a section 'IN THIS ARTICLE' with links to various topics and a 'Tweak this link' section with links to 'Subscribe', 'Ask questions', 'Follow goals', 'Report features', 'Report issues', and 'Watch videos'.



Debemos asegurarnos primero de que tenemos el JDK instalado, para ello abrimos una terminal (Tecla Windows y escribimos "**cmd**"), ahora escribimos el siguiente comando sin las comillas: "**java --version**", nos debería salir un mensaje como el siguiente:

```
java 17.0.4.1 2022-08-18 LTS
Java(TM) SE Runtime Environment (build 17.0.4.1+1-LTS-2)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.4.1+1-LTS-2, mixed mode, sharing)
```

Si no nos sale el mensaje, muy probablemente no tengamos el JDK instalado. Podemos conseguir uno de forma gratuita desde la página mencionada anteriormente. Deslizando un poco hacia abajo, encontraremos un listado de JDKs que podemos descargar, hacemos clic sobre el enlace que está señalado con la marca verde en la imagen de abajo, nos redirigirá a la página a través de la cuál podremos descargar e instalar nuestro JDK.

#### Installing a Java Development Kit (JDK)

If you have never installed a JDK before and need to install one, we recommend you to choose from one of these sources:

- [Amazon Corretto](#)
- [Azul Zulu](#)
- [Eclipse Adoptium's Temurin](#)
- [Microsoft Build of OpenJDK](#)
- [Oracle Java SE](#) ✓
- [Red Hat build of OpenJDK](#)
- [SapMachine](#)

Una vez seguros de que tenemos nuestro JDK instalado, tendremos dos vías de instalación para el editor y las herramientas necesarias para desarrollar aplicaciones con Java: podemos descargarnos el ejecutable "**Coding Pack for Java**" (disponible sólo para Windows y MacOS), a través de los botones en magenta: "**Install the Coding Pack for Java - Windows**" y "**Install the Coding Pack for Java - macOS**", y proceder con la instalación mediante esta aplicación; o podemos realizarla manualmente siguiendo los siguientes pasos:

1. Hacemos clic sobre el botón azul en la esquina superior derecha de la pantalla que dice "**Download**" y descargamos el instalador de Visual Studio Code:



Aquí escogemos el instalador dependiendo de nuestro sistema operativo: el primero es para sistemas operativos Windows, el segundo para sistemas operativos Linux y el tercero para Mac, hacemos clic sobre los recuadros en azul grandes para descargar nuestro instalador e instalamos VS Code.

- Una vez instalado Visual Studio Code, volvemos a nuestra página [VS Code Tutorial](#), e instalamos las extensiones de Java para Visual Studio Code, para ello hacemos clic sobre e botón de color magenta que dice **"Install the Extension Pack for Java"**.

### Installing extensions #

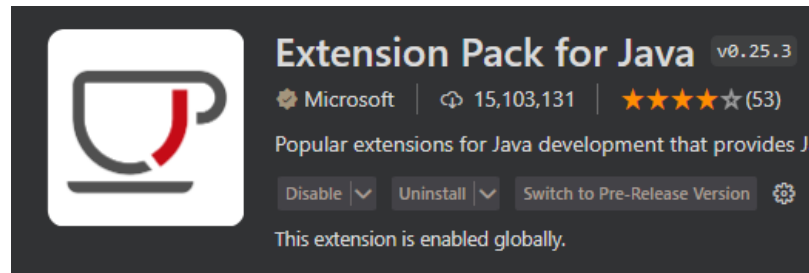
If you are an existing VS Code user, you can also add Java support by installing the [Extension Pack for Java](#), which includes these extensions:

- [Language Support for Java™ by Red Hat](#)
- [Debugger for Java](#)
- [Test Runner for Java](#)
- [Maven for Java](#)
- [Project Manager for Java](#)
- [Visual Studio IntelliCode](#)

**Install the Extension Pack for Java**

The [Extension Pack for Java](#) provides a Quick Start guide and tips for code editing and debugging. It also has a FAQ that answers some frequently asked questions. Use the command `Java: Tips for Beginners` from the Command Palette (`Ctrl+Shift+P`) to launch the guide.

- Nos abrirá una ventana de Visual Studio Code que contiene información similar a la siguiente:



Cuando se realice una instalación por primera vez, nos saldrá un botón diciendo **"Instalar"** en esta ventana, hacemos clic sobre él y esperamos a que se acaben de instalar las extensiones de Java.

4. Finalizado el paso anterior ya estaremos listos para crear nuestro programa en Java y ejecutarlo.

## Bibliografía

- [1] Wikipedia, «Wikipedia - Python,» [En línea]. [Último acceso: 1 Octubre 2022].
- [2] Python, «Python - About,» [En línea]. Available: <https://www.python.org/about/>. [Último acceso: 1 Octubre 2022].
- [3] Python, «Python - Beguiners guide,» 15 Abril 2022. [En línea]. Available: <https://wiki.python.org/moin/BeginnersGuide/Download>. [Último acceso: 1 Octubre 2022].
- [4] Wikipedia, «Wikipedia - Java (Lenguaje de programación),» 27 Septiembre 2022. [En línea]. Available: [https://es.wikipedia.org/wiki/Java\\_\(lenguaje\\_de\\_programación\)](https://es.wikipedia.org/wiki/Java_(lenguaje_de_programación)). [Último acceso: 1 Octubre 2022].
- [5] Chuidiang, «Ejemplos java y C/linux,» [En línea]. Available: <https://www.chuidiang.org/java/classpath/inicios.php>. [Último acceso: 1 Octubre 2022].