



UNIDAD 1

Introducción a Python

Unidad 1: Introducción a Python

Programa, aplicación o software.

Conjunto de instrucciones elaboradas para ordenar a un dispositivo digital ejecutar alguna tarea.

```
Dim i, j, Array_Used As Integer
Dim MyArray() As String
Dim InBuffer, Temp As String
Array_Used = 0
ReDim MyArray(50)
' Abrir archivo de texto . . .
Do While Not EOF(file_no)
    Line Input #file_no, MyArray(Array_Used)
    Array_Used = Array_Used + 1
    If Array_Used = UBound(MyArray) Then
        ReDim Preserve MyArray (UBound(MyArray) + 50)
    End If
```

Lenguaje de Programación.

Son herramientas que permiten desarrollar programas, encargados de definir el comportamiento de los dispositivos físicos y lógicos

*PYPL PopularitY of
Programming
Language*

[https://pypl.github.io/
PYPL.html](https://pypl.github.io/PYPL.html)

Worldwide, Sept 2021			
Rank		Language	Share
1		Python	29.48 %
2		Java	17.18 %
3		JavaScript	9.14 %
4		C#	6.94 %
5		PHP	6.49 %
6		C/C++	6.49 %
7		R	3.59 %
8		TypeScript	2.18 %
9		Swift	2.1 %
10		Objective-C	2.06 %

Se conforma de símbolos, reglas de sintaxis y semántica que definen su estructura principal y le dan un significado a sus elementos y expresiones.

Unidad 1: Introducción a Python

Nivel de abstracción.



Ser
humano

ALTO NIVEL

Perl, PHP, Visual Basic
C, C++, Fortran, BASIC
Assembler
Código de máquina

BAJO NIVEL



Dispositivo
digital

```
package main  
  
import "fmt"  
  
func main() {  
    fmt.Printf("hello, world")  
}
```

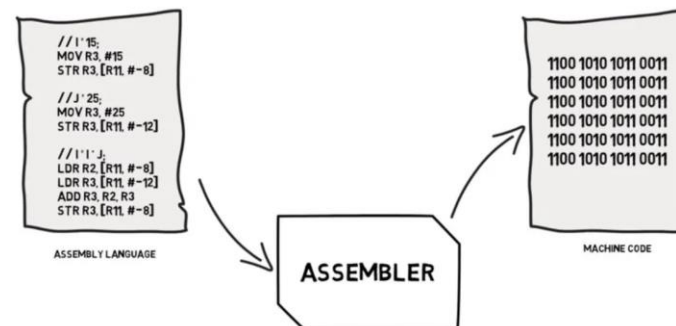
Lenguaje de alto nivel



```
0101010111101110001101  
0100010100010101001010  
0101010010101010000101  
0011010001010100011110  
0110010100101010101001  
1110001101010010010001
```

Lenguaje de máquina

Los lenguajes de "alto nivel" son mas comprensibles para los programadores aunque suelen tener menor rendimiento.



Por el contrario, los lenguajes de bajo nivel son mas complejos pero aprovechan mejor los recursos de los dispositivos.

```
01101000 01100101  
01101100 01101100  
01101111 00100000  
01101111 01101111  
01110010 01101100  
01100100
```

Lenguaje Máquina



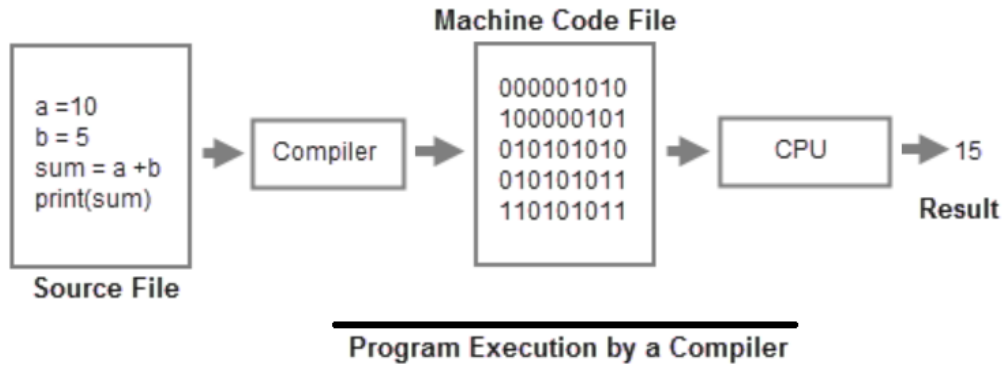
Procesador

Hardware

Unidad 1: Introducción a Python

Tipos de lenguajes de alto nivel

1. Lenguajes compilados

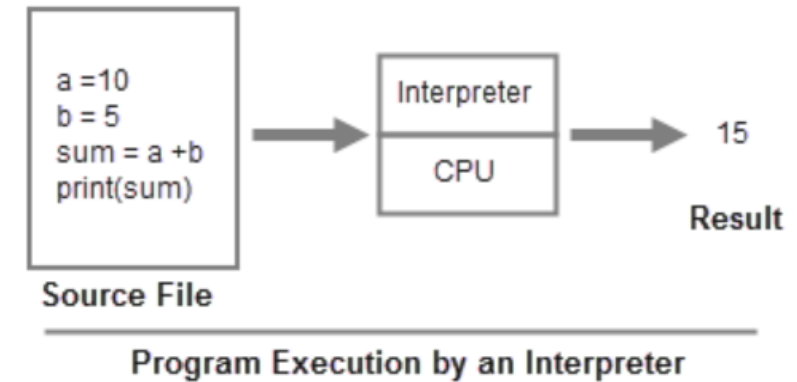


Aquellos que son traducidos al lenguaje de máquina de una sola vez

*C, C++, Ada,
ALGOL, Delphi, Fortran, Java,
Lisp, Pascal, Perl, Go, Scala,
Swift, Rust*

Tienen un rendimiento superior y están optimizados para su ejecución.

2. Lenguajes interpretados



Son traducidos, instrucción por instrucción, al lenguaje máquina.

Ruby, Python, JavaScript, PHP, VBScript

No dependen de la máquina donde se ejecuta el intérprete

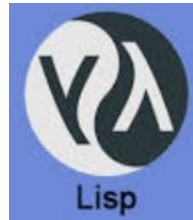
Unidad 1: Introducción a Python

Paradigmas de programación

- **Imperativa:** Órdenes directas al dispositivo, que van cambiando el estado del programa.



- **Declarativa:** Descripción de los resultados esperados sin especificar explícitamente los pasos para alcanzarlos.



- **Estructurada:** Es un caso particular de programación procedural, la diferencia es que hay un flujo jerárquico con combinación de elementos como secuencias, selección y repetición.



- **Orientada a objetos:** Se usan estructuras de datos que tienen datos y métodos (objetos) con sus interacciones para diseñar programas.



Unidad 1: Introducción a Python

Términos generales.

Compilador: Programa que realiza un proceso de compilación (traducción del lenguaje fuente a lenguaje máquina) de un programa escrito en un lenguaje de programación de alto nivel.

Depurador: Herramienta para ayudar a la localización de errores de un programa. Puede establecer puntos de interrupción, parada simple a través de un programa e inspecciona el estado de las variables.

IDE: Integrated Development Environment, es un entorno de programación, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI). Pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes.

IDLE: Integrated Development and Learning Environment, el entorno integrado de desarrollo de Python, es un editor básico y un entorno de intérprete que se incluye con la distribución estándar de Python.

Sintaxis: Conjunto de reglas que deben seguirse al escribir el código de los programas para considerarse como correctos.

Código fuente: Las instrucciones escritas en algún lenguaje de programación, pero que no es directamente ejecutable por la computadora, sino que debe ser traducido a otro lenguaje o código binario

Unidad 1: Introducción a Python

Python - Definición

Es un lenguaje de programación interpretado y orientado a objetos, que ha ganado popularidad debido a su clara sintaxis y legibilidad. Es relativamente fácil de aprender y portátil, lo que significa que su código se pueden interpretar en varios sistemas operativos.



Python - Creación

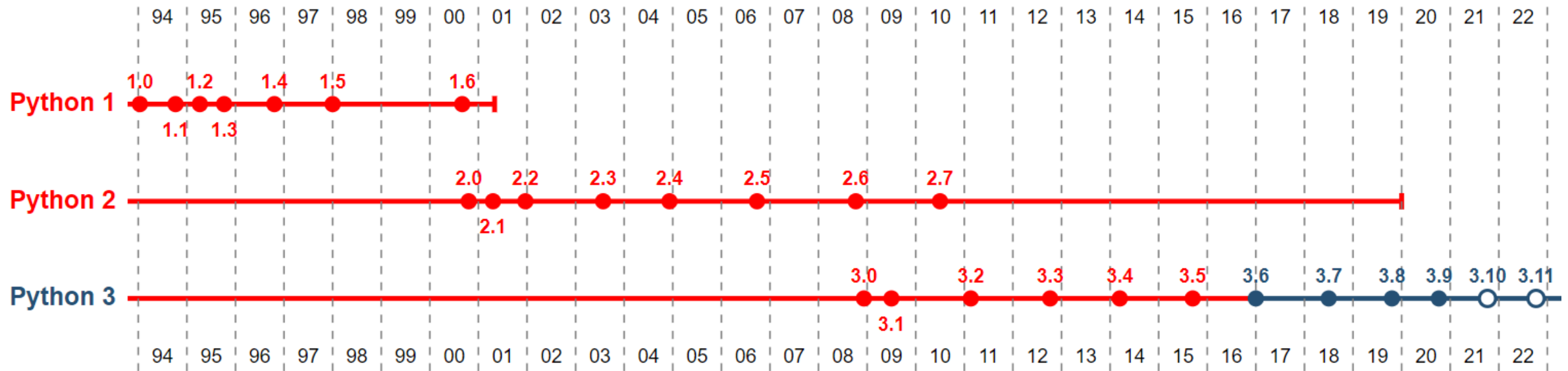
Python fue creado a finales de los ochenta por Guido van Rossum en el instituto holandés Centrum Wiskunde e Informatica (CWI). En la actualidad Python se desarrolla como un proyecto de Código abierto, administrado por la “Python Software Foundation”.



Unidad 1: Introducción a Python

Python - Versiones

Se identifican por tres números X.Y.Z:



- X corresponde a las grandes versiones de Python (1, 2 y 3), incompatibles entre sí.
- Y corresponde a versiones importantes en las que se introducen novedades en el lenguaje pero manteniendo la compatibilidad.
- Z corresponde a versiones menores que se publican durante el período de mantenimiento, en las que sólo se corrigen errores durante el primer año y fallos de seguridad.

Unidad 1: Introducción a Python

Python - Características

- Lenguaje de propósito general.
- Es multiparadigma
- Es de tipado dinámico
- Interpretado
- Es orientado a objetos
- Multiplataforma.
- Libre y gratuito
- Cantidad de bibliotecas con funciones extras.
- Comunidad

“El nombre del lenguaje proviene de la afición de su creador original, por los humoristas británicos Monty Python.”

Python - Filosofía, Zen, Mantra

- Hermoso es mejor que feo
- Explícito es mejor que implícito
- Simple es mejor que complejo
- Complejo es mejor que complicado
- Sencillo es mejor que anidado
- Escaso es mejor que denso
- La legibilidad cuenta
- Los casos especiales no son lo suficientemente especiales para romper las reglas
- Lo práctico le gana a la pureza
- Los errores no deben quedar en silencio a menos que sean silenciados
- En caso de ambigüedad, rechaza la tentación de adivinar
- Debe haber una – y preferiblemente sólo una – manera obvia de hacerlo
- Aunque esa manera puede no ser obvia en un primer momento a menos que seas holandés
- Ahora es mejor que nunca
- Aunque “nunca” es a menudo mejor que “ahora mismo”
- Si la aplicación es difícil de explicar, es una mala idea
- Si la aplicación es fácil de explicar, puede ser una buena idea
- Los espacios de nombres son una gran idea ¡hay que hacer más de eso!

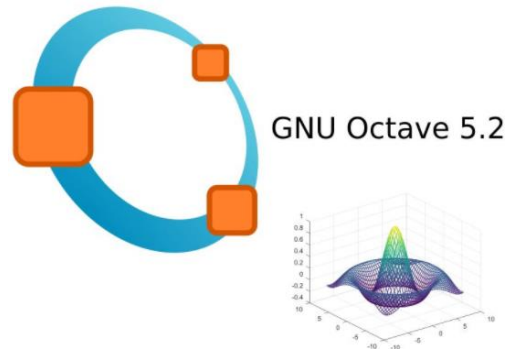
Unidad 1: Introducción a Python

Aplicaciones en la Ingeniería.

MATLAB (MATrix LABoratory) es todo un sistema de cómputo numérico que ofrece un entorno de desarrollo integrado con un lenguaje de programación propio conocido como “lenguaje M”.



Muy usado en universidades, centros de investigación y desarrollo. El licenciamiento comercial orientado a aplicación general ronda los 1000 dólares, sin considerar sus adicionales.

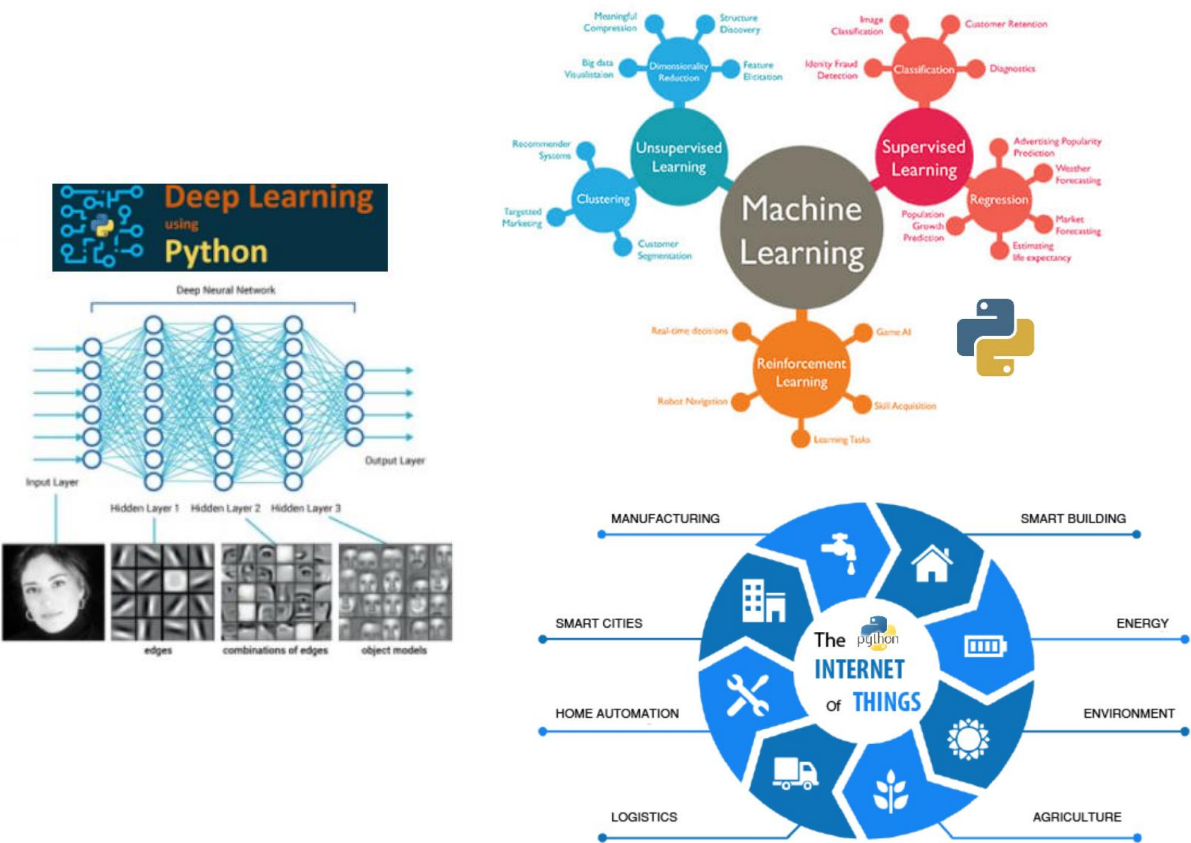


Es un programa y lenguaje de programación para realizar cálculos numéricos. Posee herramientas de impresión de gráficos y visualización incorporadas. Está considerado el equivalente libre de MATLAB, es altamente compatible.

Unidad 1: Introducción a Python

Python en la Ingeniería.

Áreas de conocimiento como el cálculo numérico, el modelado y la optimización, la extracción, gestión, manipulación y visualización de datos así como la conexión e interacción con todo tipo de entidades físicas o virtuales enlazan directamente con aplicaciones en la empresa, industria y mercado como:



Unidad 1: Introducción a Python

Naturaleza

MATLAB es de código cerrado y un producto comercial patentado. Si MathWorks alguna vez dejara de funcionar, perdería su importancia industrial.

Python es de código abierto, totalmente gratuito. Tiene una base de seguidores y usuarios mayor. Se ha vuelto popular en múltiples aplicaciones.

Sintaxis

MATLAB trata todo como una matriz, las cadenas pueden ser matrices de cadenas o matrices de caracteres; un comentario en Matlab (%).

Python trata todo como un objeto general, las cadenas se indican mediante un objeto único conocido como "str"; un comentario en Python (#).

Python vs Matlab.

IDE

MATLAB tiene un entorno de desarrollo integrador. Es una interfaz ordenada con una consola ubicada en el centro, y un explorador de variables

Python no adopta un entorno de desarrollo predeterminado. Spyder y JupyterLab funcionan con la misma eficacia que el IDE de MATLAB.

Las herramientas

MATLAB no tiene muchas bibliotecas, su biblioteca habitual contiene kits de herramientas incorporados para cubrir los desafíos científicos y computacionales avanzados.

Python, todas sus bibliotecas incluyen muchos módulos útiles para diversas aplicaciones. Entre las mejores bibliotecas de Python se encuentran NumPy, SciPy, PyTorch, OpenCV Python, Keras, TensorFlow, Matplotlib, Theano, Requests y NLTK.

Unidad 1: Introducción a Python

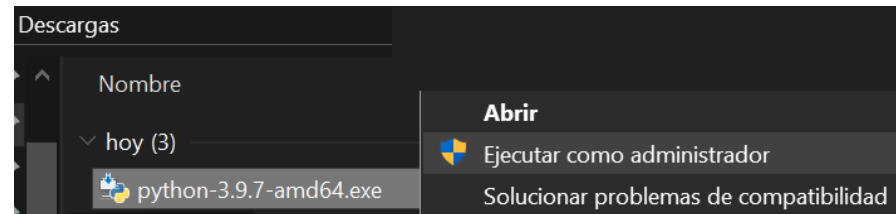
Instalación de Python.

Tomando en cuenta que Python es multiplataforma la instalación puede realizarse en múltiples sistemas operativos. Para MS Windows se realiza el siguiente procedimiento:

1. Acceder mediante un navegador web a la dirección: <https://www.python.org/>, en la sección de descargas buscar la última versión disponible. Esto descargará la versión de 64 bits.

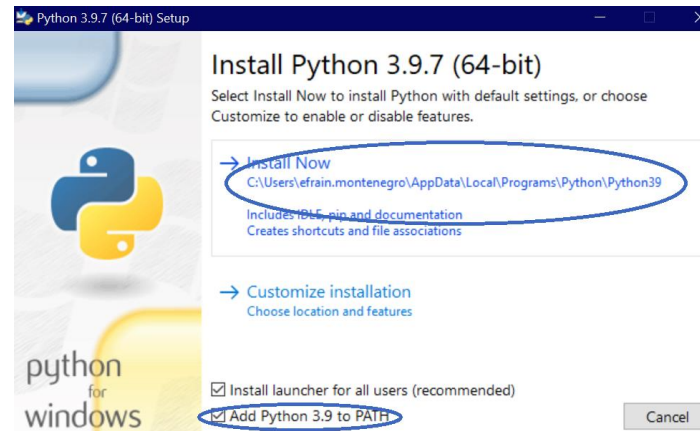


2. Ejecutar el instalador como administrador dentro de MS Windows.

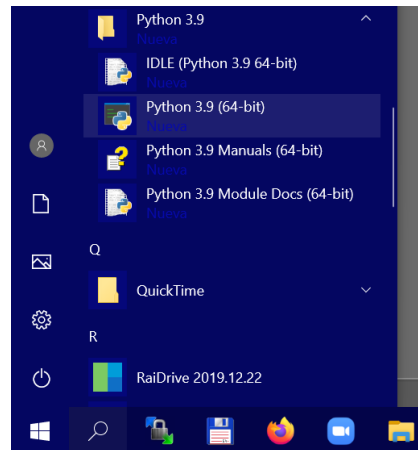


Unidad 1: Introducción a Python

- Tomar en cuenta la ruta de la instalación en caso de necesitar alguna configuración adicional, marcar la opción "Add Python 3.9 to PATH" e iniciar la instalación .



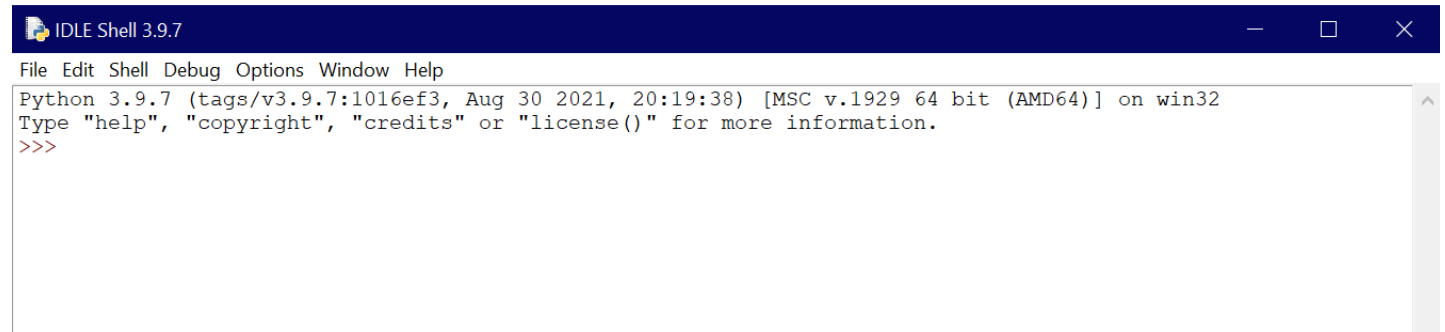
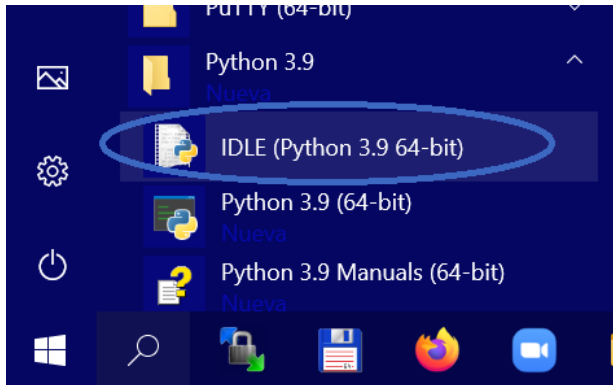
- El resultado de la instalación está disponible en el menú inicio.



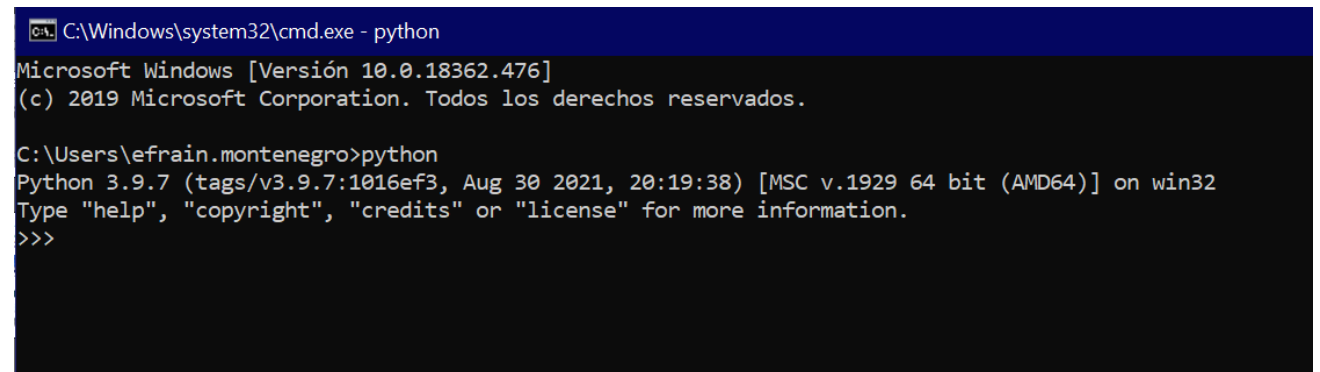
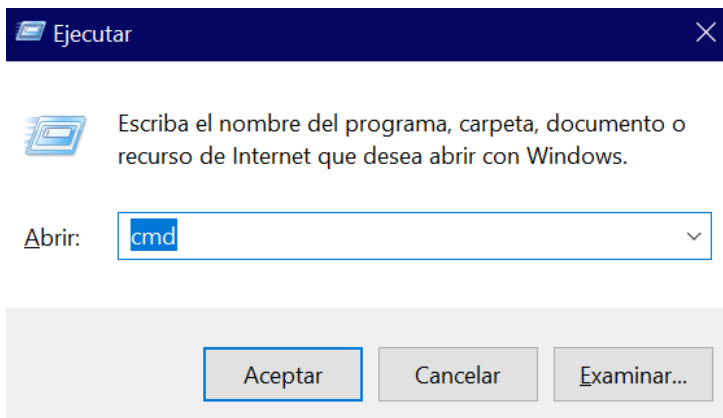
Unidad 1: Introducción a Python

El shell de Python.

1. Iniciar el Shell, consola o IDLE nativo de Python desde el menú inicio de MS Windows.



2. Iniciar el IDLE mediante el comando python desde el cmd de MS Windows.



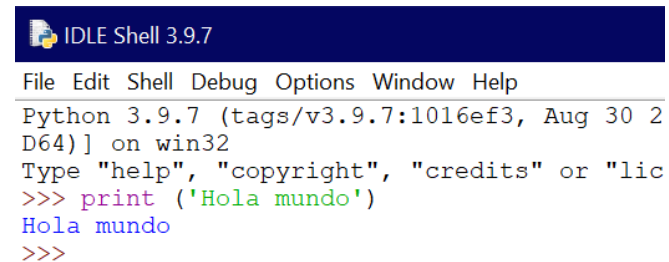
Unidad 1: Introducción a Python

El shell de Python.

El uso de la consola de IDLE en Python implica el desarrollo de código fuente con la sintaxis asociada, entre las tareas más generales que se pueden implementar está:

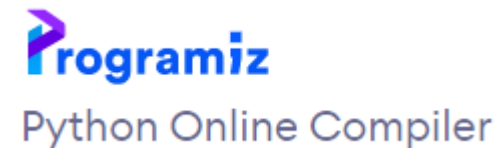
- Ejecutar múltiples instrucciones sobre la consola:

```
import this  
print("Hola Mundo")  
# crear comentarios
```



```
IDLE Shell 3.9.7  
File Edit Shell Debug Options Window Help  
Python 3.9.7 (tags/v3.9.7:1016ef3, Aug 30 2020) on win32  
Type "help", "copyright", "credits" or "license()" for more  
>>> print('Hola mundo')  
Hola mundo  
>>>
```

- Guardar archivos con extensión “.py”, scripts de código sencillo.
- Ejecutar los archivos, scripts sencillos desde el IDLE.
- Probar alternativas de implementación del IDLE sobre internet.



Unidad 1: Introducción a Python

Anaconda - Definición

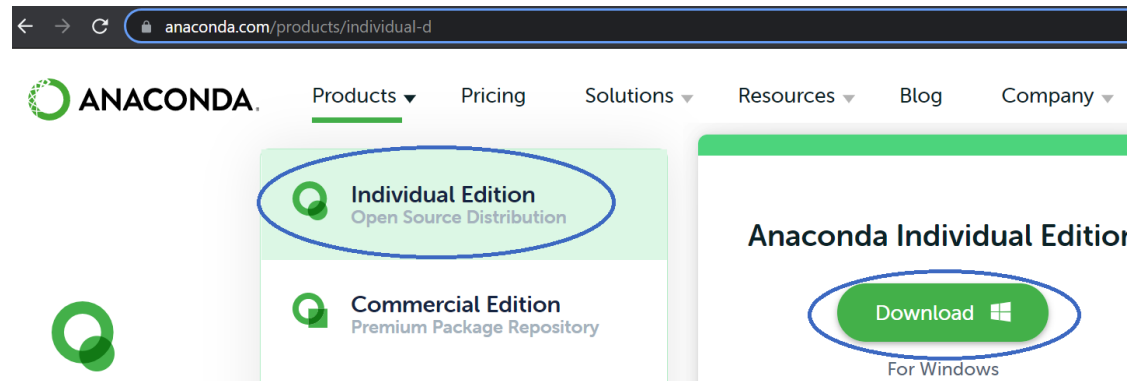
Conda es un gestor de paquetes y un sistema de gestión de entornos de código abierto y multiplataforma.



La versión comercial del gestor, orientada a propósito general, y motivada en el análisis de datos, se conoce como Anaconda, y es gestionada por la empresa “Continuum Analytics”.

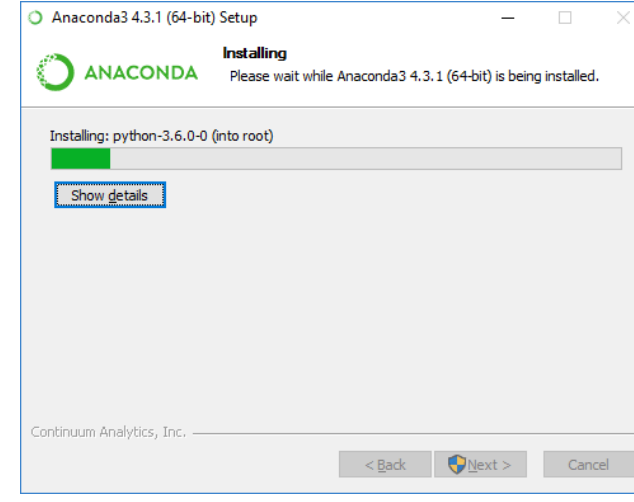
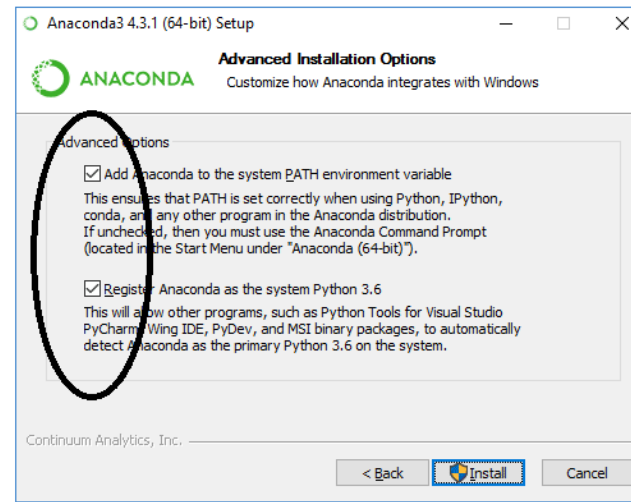
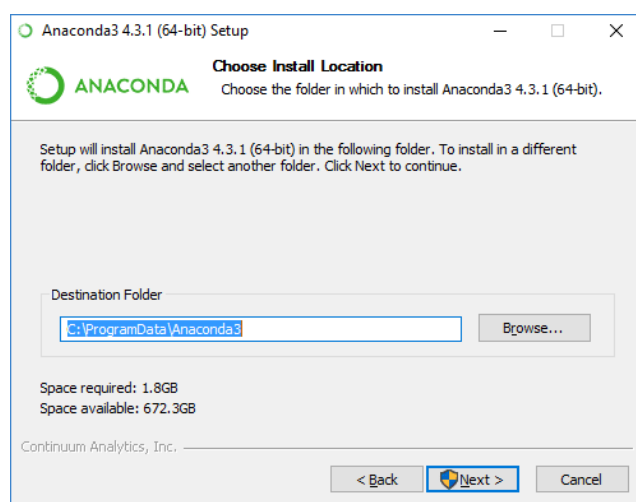
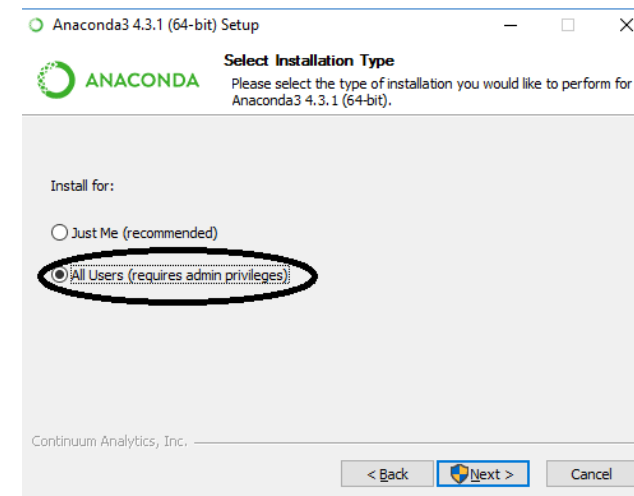
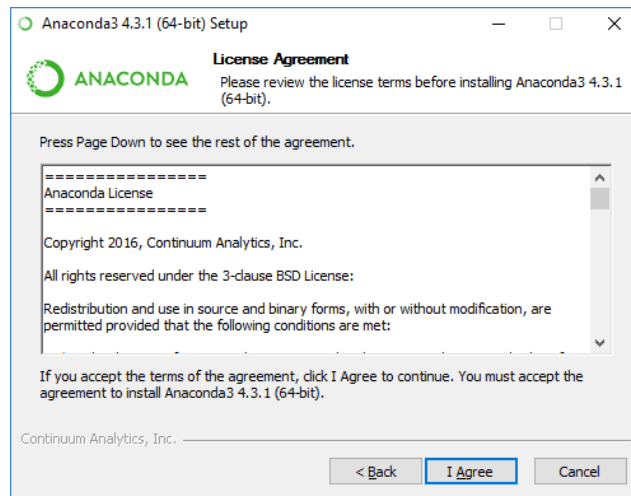
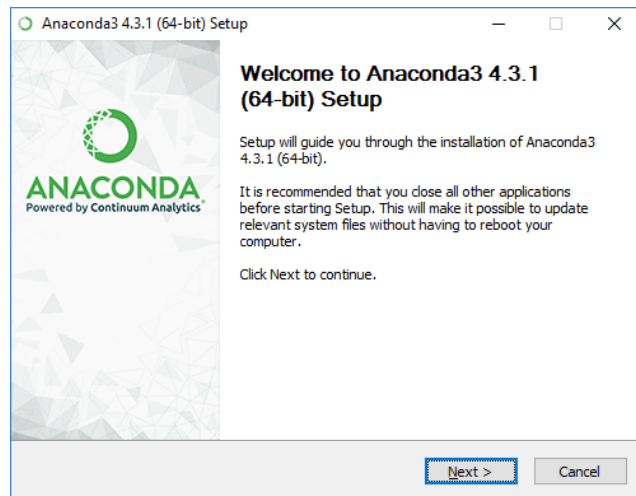
Anaconda – Instalación.

1. Acceder a la página web oficial <https://www.anaconda.com/> y descarga Anaconda, la versión de uso individual.



Unidad 1: Introducción a Python

- Una vez descargado ejecuta el instalador como administrador, avanzar en la instalación tomando en cuenta lo siguiente:



Unidad 1: Introducción a Python

Jupyter Notebook - IPython

IPython es un shell o consola interactivo que añade funcionalidades extra al modo interactivo incluido con Python, como resaltado de líneas y errores mediante colores, una sintaxis adicional para el shell, autocompletado mediante tabulador de variables, módulos y atributos; entre otras funcionalidades.



IP[y]:

Jupyter Notebook es una aplicación web cliente-servidor lanzada en 2015 como una mejora de IPython. Permite crear y compartir documentos que siguen un esquema versionado. Puede manejar, entre otras cosas, código, texto, fórmulas matemáticas, ecuaciones, o también contenido multimedia.

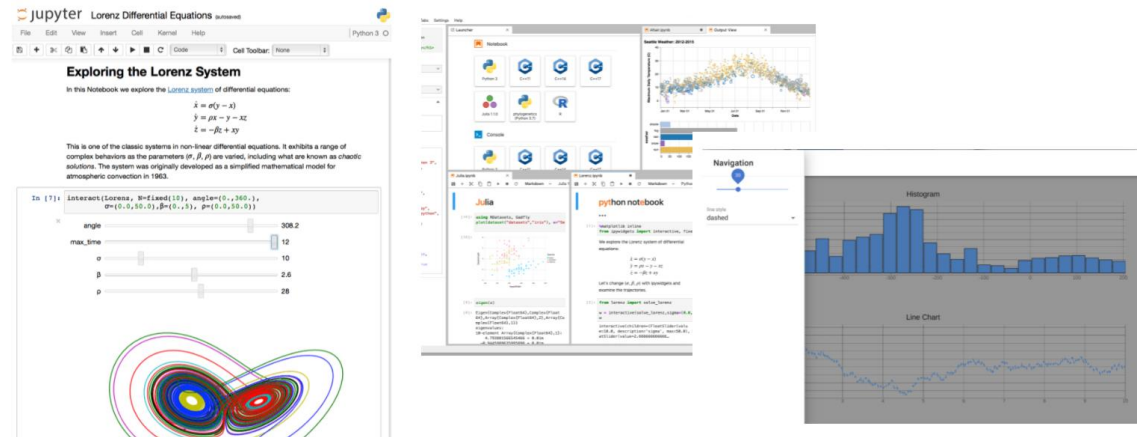


Es posible trabajar también con otros lenguajes como C++, R, Julia, Ruby, JavaScript, PHP o Java.

Unidad 1: Introducción a Python

Jupyter Notebook - Ventajas

- Escribir, editar, copiar y pegar bloques de código fácilmente.
- El autocompletado facilita el acceso a los nombres de lo que se está usando.
- Permite realizar anotaciones en el código con enlaces, texto de diferentes tamaños, viñetas, etc. para hacer la información más accesible.
- Permite mostrar gráficos junto al código.



Jupyter Notebook - Principales Usos

- Depuración de datos, distinguir entre los datos que son importantes y los que no lo son al ejecutar un análisis de “big data”.
- Modelización estadística, estimación de probabilidad de distribución de una características específicas.
- Creación y entrenamiento de modelos de aprendizaje automático, diseño, programación y entrenamiento.
- Representación gráfica de datos para visualizar con claridad patrones, tendencias, interdependencias, etc.

Unidad 1: Introducción a Python

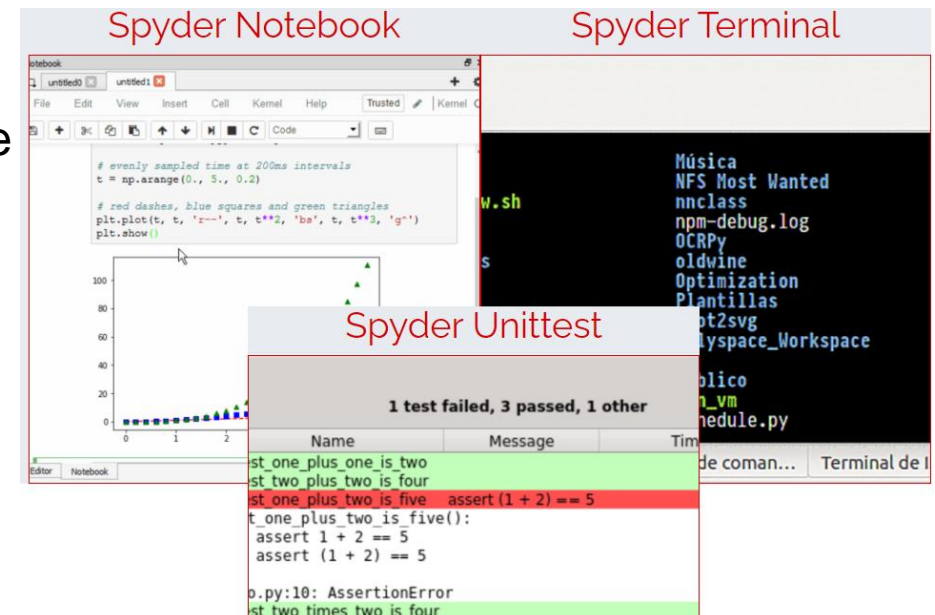
Spyder - IDE

Scientific Python Development Environment, es un entorno de desarrollo integrado y multiplataforma de código abierto para programación científica en Python.



Spyder - características

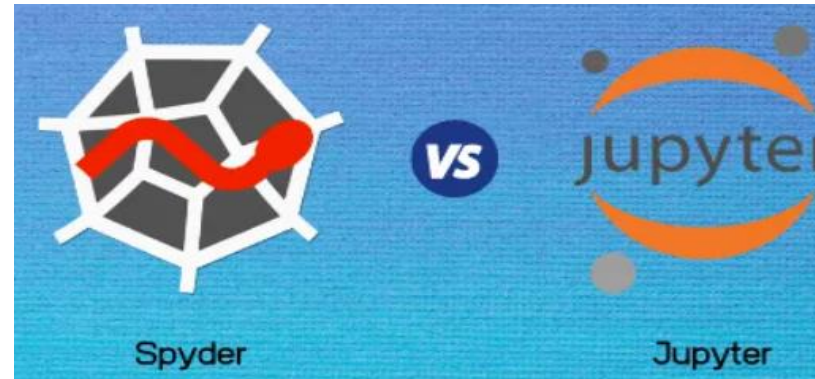
- El editor integrado es multilenguaje.
- Posee un navegador de función/clase, funciones de análisis de código.
- Consola interactiva.
- Posee un visor de documentación integrado.
- Explorador de variables creadas durante la ejecución de un archivo.
- Soporte de expresiones regulares.
- Explorador de archivos para una mayor comodidad.
- Acceso al registro del historial.



Unidad 1: Introducción a Python

Jupyter y Spyder

Jupyter es una muy popular y utilizado para el análisis de datos.. Puede ejecutar cada bloque de código por separado. Por ejemplo, se puede imprimir un gráfico usando librerías. Crear nuevos bloques de código e imprimir otro gráfico. También hay funciones interesantes como % timeit que prueban la velocidad de ejecución del código.



Spyder es únicamente un entorno de desarrollo integrado para python como atom, visual studio, etc. Es más fácil de aprender y comenzar a ejecutar código. También es muy popular, especialmente en academia.

Se prefiere usar Jupyter notebook para analizar datos en tiempo real. Al desarrollar un programa o implementar un nuevo código en sobre datos analizados, se prefiere el uso de Spyder.



ESCUELA
POLITÉCNICA
NACIONAL



ESCUELA
POLITÉCNICA
NACIONAL



python™