



Programación Python

José Javier Galán Hernández:



Programación Python

1. Introducción a la programación

1 Introducción a la programación

Tradicionalmente podemos definir la programación como la automatización de tareas, especialmente las que suponen un mayor esfuerzo para el ser humano o aquellas que por ser repetitivas no aportan valor y se pueden realizar desatendida.

En la ciencia de datos la programación adquiere un valor extra, además del automatismo implican la codificación del código necesario para representar de forma visual el resultado de aplicar IA a un conjunto enorme de datos, hablamos de big data y machine learning.



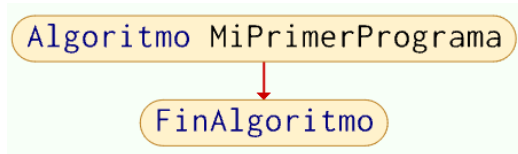
Con este objeto surgen lenguajes como R y Python muy enfocados a obtener estos resultados pero que en esencia son lenguajes de programación, por lo tanto es necesario tener al menos unas destrezas mínimas programando.



1 Introducción a la programación

Pseudocódigo

El pseudocódigo es independiente del lenguaje de programación, representaremos mediante símbolos lo que queremos conseguir, en la programación estructurada el pseudocódigo seguirá un orden secuencial y lógico. A continuación vemos la composición básica de un programa, todo programa debe tener un comienzo y un final. Dicho programa representa la solución de un problema y esto es lo que se conoce como algoritmo.



Algoritmo MiPrimerPrograma

FinAlgoritmo

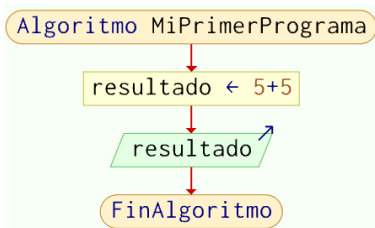


1 Introducción a la programación

Pseudocódigo

Un algoritmo puede tener tantas acciones, conocidas como instrucciones, dentro de codificación como sean necesarias. Para conseguir el éxito del objetivo propuesto deberemos utilizar todas las instrucciones a nuestro alcance en el orden secuencial correcto con la mayor eficiencia posible.

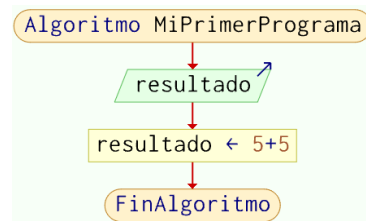
Orden correcto:



```
Algoritmo MiPrimerPrograma
    resultado<-5+5
    Escribir resultado
FinAlgoritmo
```

Resultado de la ejecución: 10

Orden incorrecto:



```
Algoritmo MiPrimerPrograma
    Escribir resultado
    resultado<-5+5
FinAlgoritmo
```

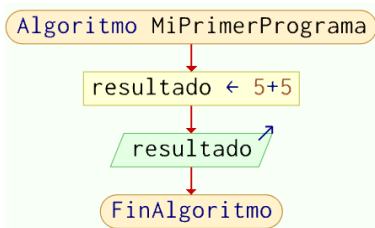
Resultado de la ejecución: 0

1 Introducción a la programación

Pseudocódigo

Un algoritmo puede tener tantas acciones, conocidas como instrucciones, dentro de codificación como sean necesarias. Para conseguir el éxito del objetivo propuesto deberemos utilizar todas las instrucciones a nuestro alcance en el orden secuencial correcto con la mayor eficiencia posible.

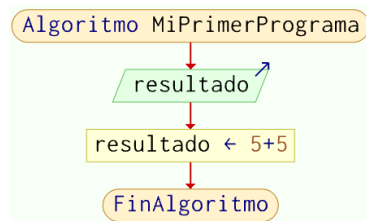
Orden correcto:



```
Algoritmo MiPrimerPrograma
    resultado ← 5+5
    Escribir resultado
FinAlgoritmo
```

Resultado de la ejecución: 10

Orden incorrecto:



```
Algoritmo MiPrimerPrograma
    Escribir resultado
    resultado ← 5+5
FinAlgoritmo
```

Resultado de la ejecución: 0



Programación Python

2. PYTHON

2 Python



Python es un lenguaje de programación multiparadigma. Esto significa que más que forzar a los programadores a adoptar un estilo particular de programación, permite varios estilos: programación orientada a objetos, programación imperativa y programación funcional.

Es un lenguaje interpretado, de alto nivel y enfocado principalmente a la legibilidad y facilidad de uso

Los lenguajes interpretados se pueden ejecutar sin compilar usando un intérprete, no tienen código binario y su código se llama script. Al disponer de intérprete no dependen de la plataforma y suelen ser más lentos que los compilados.

2 Python

Python destaca por el elevado uso de bibliotecas y una gran cantidad de paquetes (pilas incluidas), ser software libre (licencia “Python Software Foundation License”) distribuyéndose gratuitamente, sin pagar licencias.

Su creador, Guido van Rossum lo define como código limpio y legible. Simple sin ser limitado.



2 Python

Filosofía Python

- Hermoso es mejor que feo.
- Explícito es mejor que implícito.
- Simple es mejor que complejo.
- Complejo es mejor que complicado.
- Plano es mejor que anidado.
- Escaso es mejor que denso.
- La legibilidad cuenta.
- Los casos especiales no son lo suficientemente especiales como para romper las reglas.
- Aunque la practicidad supera la pureza.
- Los errores nunca deben pasar en silencio.
- A menos que sea silenciado explícitamente.
- Ante la ambigüedad, rechaza la tentación de adivinar.
- Debe haber una, y preferiblemente solo una, forma obvia de hacerlo.
- Aunque esa manera puede no ser obvia al principio a menos que seas holandés.
- Ahora es mejor que nunca.
- Aunque nunca es mejor que el ahora correcto.
- Si la implementación es difícil de explicar, es una mala idea.
- Si la implementación es fácil de explicar, puede ser una buena idea.
- Los espacios de nombres son una gran idea, ¡hagamos más de eso!

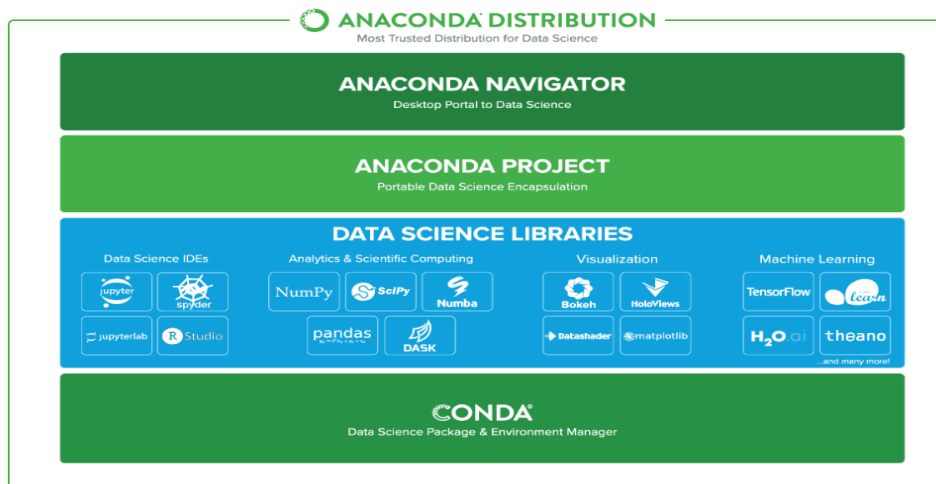


Programación Python

3. Instalación y primer contacto con el entorno de trabajo: Anaconda y Jupyter

3. Instalación y primer contacto con el entorno de trabajo: Anaconda y Jupyter

Anaconda es una Suite de código abierto que abarca una serie de aplicaciones, librerías y conceptos diseñados para el desarrollo de la Ciencia de datos como Python. En líneas generales Anaconda Distribution es una distribución que funciona como un gestor de entorno, un gestor de paquetes y que posee una colección de más de 720 paquetes de código abierto.




3. Instalación y primer contacto con el entorno de trabajo: Anaconda y Jupyter

Accedemos a la url <https://www.anaconda.com/> y hacemos click en Download para instalar **ANACONDA**

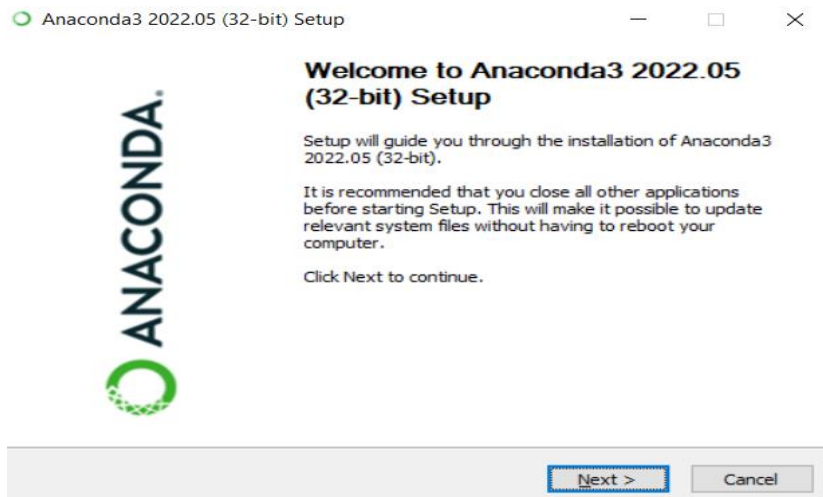


3. Instalación y primer contacto con el entorno de trabajo: Anaconda y Jupyter

Se ha descargado el ejecutable de instalación:

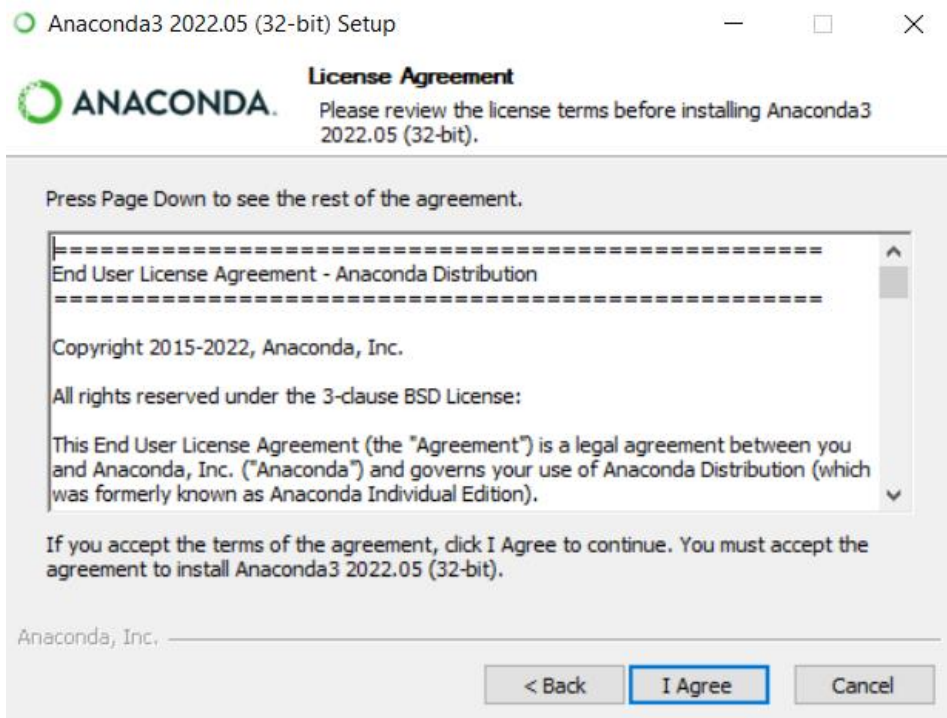
 Anaconda3-2022.05-Windows-x86.exe

Hacemos doble click en el fichero descargado y hacemos click en **Next**.



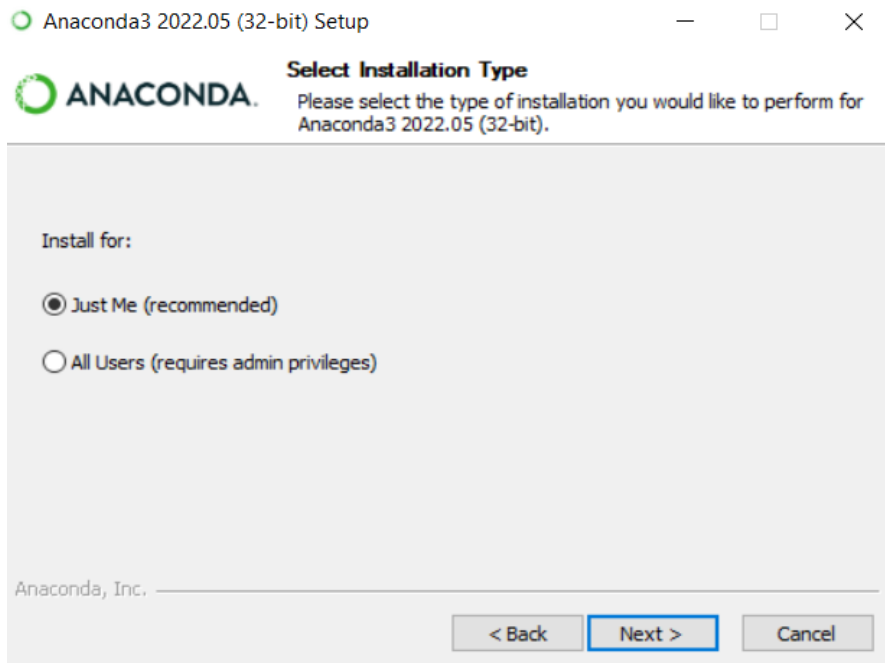
3. Instalación y primer contacto con el entorno de trabajo: Anaconda y Jupyter

Si estamos de acuerdo con el acuerdo de licencia hacemos click en **I agree**



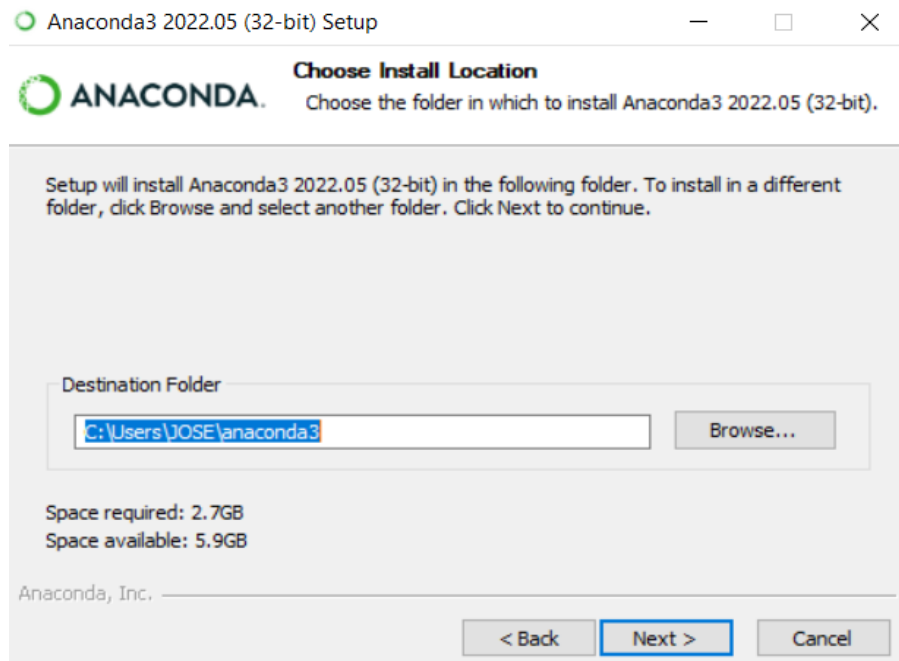
3. Instalación y primer contacto con el entorno de trabajo: Anaconda y Jupyter

Seleccionamos la opción solo para nuestro usuario y pulsamos **Next**



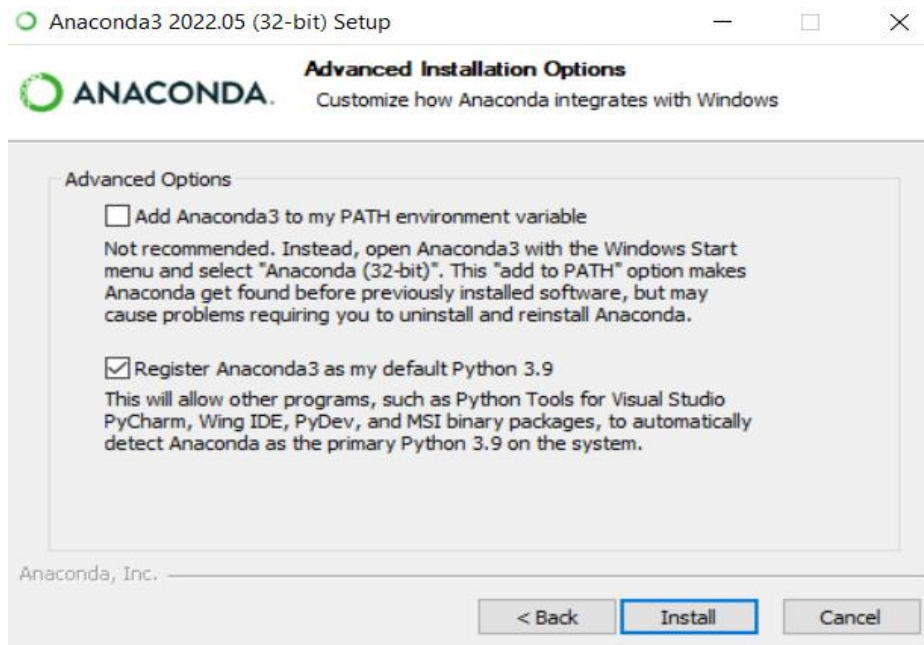
3. Instalación y primer contacto con el entorno de trabajo: Anaconda y Jupyter

Ahora indicamos la ruta donde se copiarán los archivos de instalación, podemos dejar la ruta por defecto y pulsar **Next**



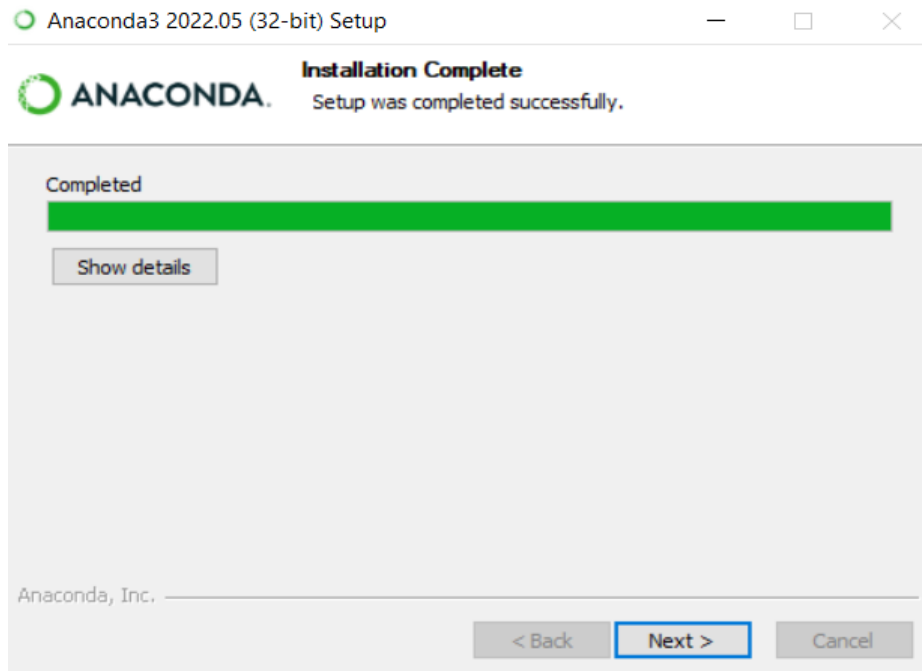
3. Instalación y primer contacto con el entorno de trabajo: Anaconda y Jupyter

Finalmente pulsamos **Install**



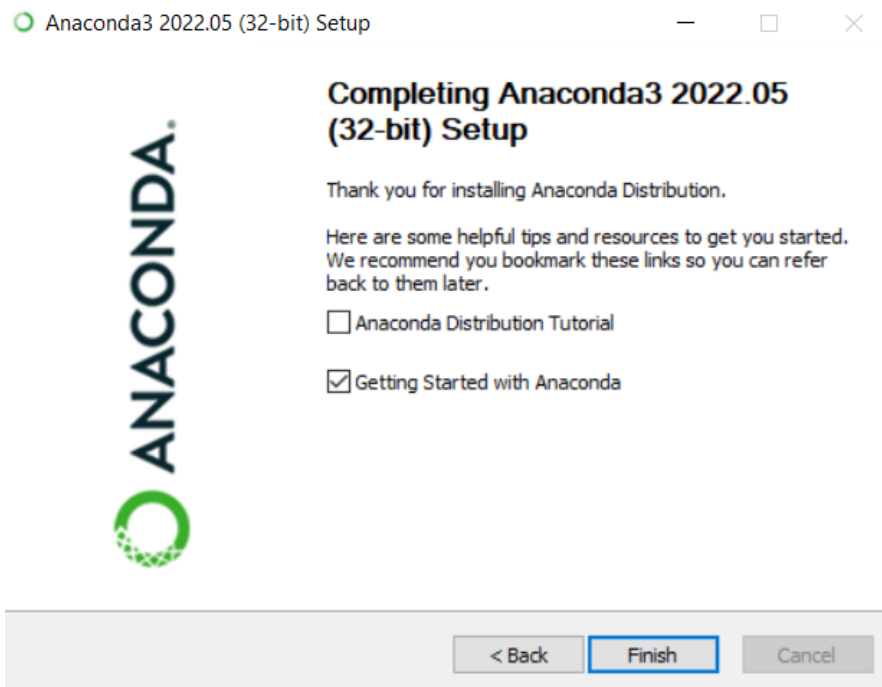
3. Instalación y primer contacto con el entorno de trabajo: Anaconda y Jupyter

Tras unos minutos la instalación termina.



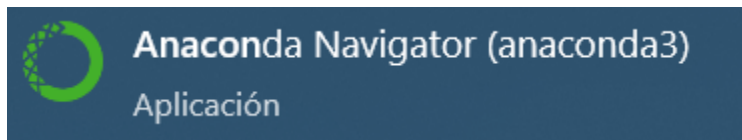
3. Instalación y primer contacto con el entorno de trabajo: Anaconda y Jupyter

Anaconda esta instalado, pulsamos Finish.

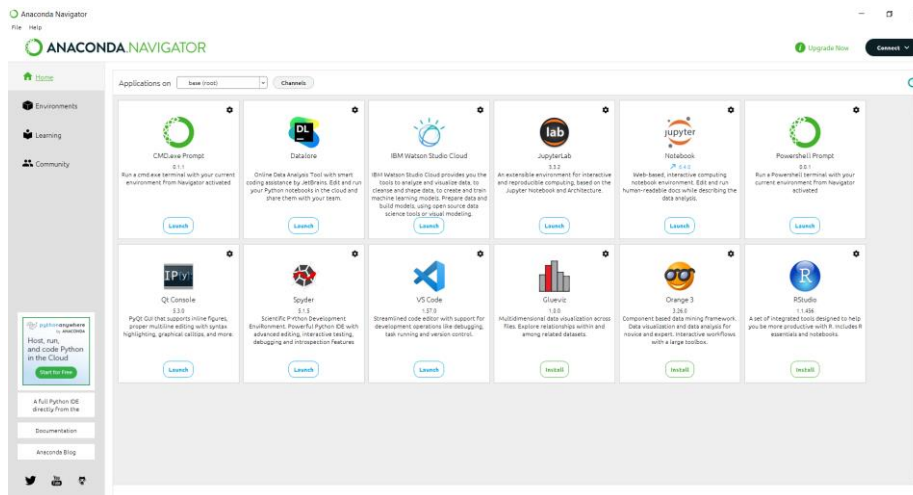


3. Instalación y primer contacto con el entorno de trabajo: Anaconda y Jupyter

Podemos abrirlo haciendo click en el acceso directo



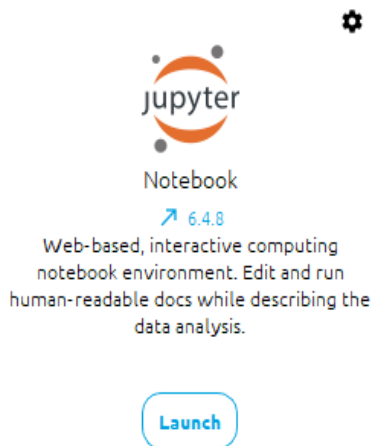
Viendo la pantalla principal



3. Instalación y primer contacto con el entorno de trabajo: Anaconda y Jupyter

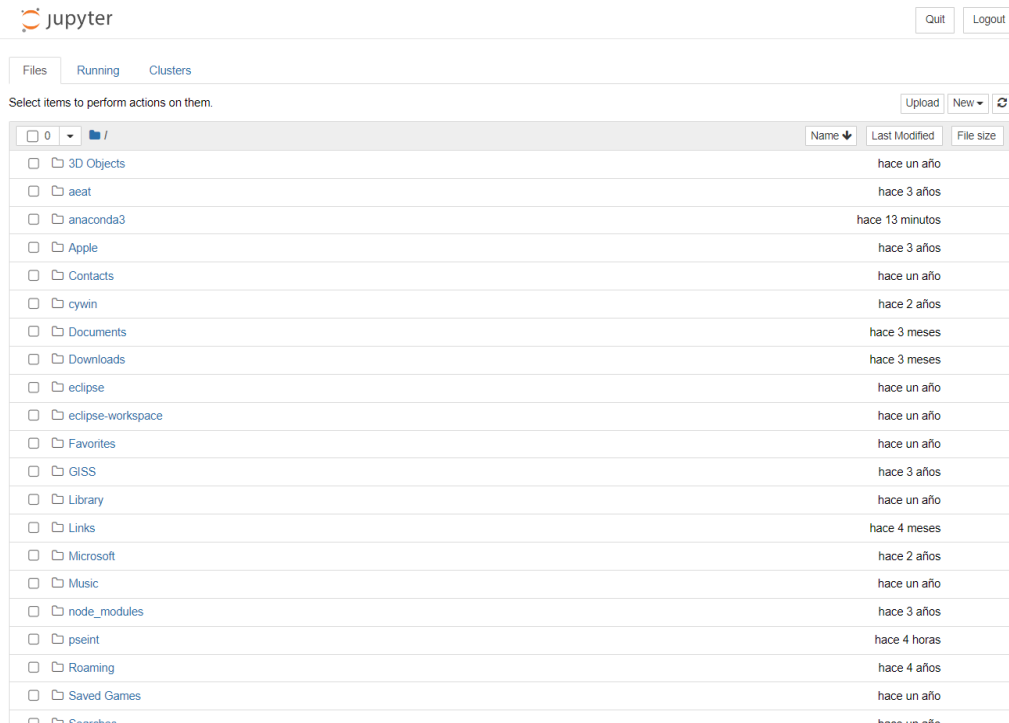
Jupyter es una aplicación web que se puede abrir desde cualquier navegador permitiendo crear documentos que contienen código y pueden ser interpretados como el lenguaje de programación Python.

ABRIMOS JUPYTER. En el menú principal de Anaconda hacemos click en **Launch** de la aplicación Jupyter



3. Instalación y primer contacto con el entorno de trabajo: Anaconda y Jupyter

Esto abre en el navegador el entorno Jupyter donde podemos desarrollar nuestros proyectos.

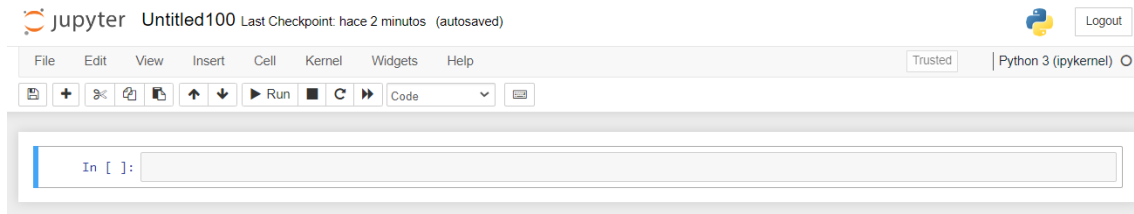


3. Instalación y primer contacto con el entorno de trabajo: Anaconda y Jupyter

Para comenzar hacemos click en **New / Python 3**

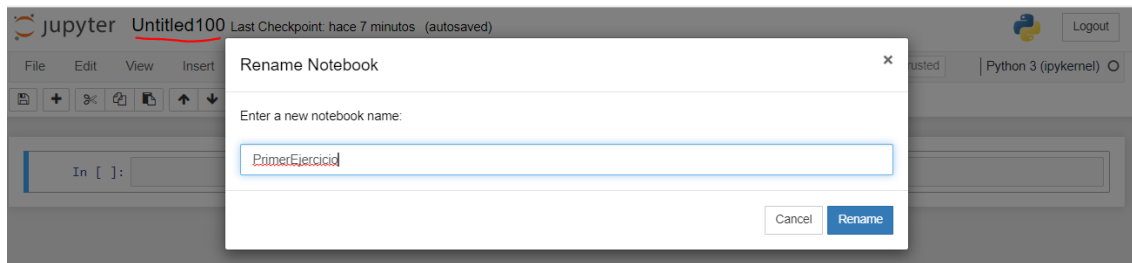


Con esto hemos creado nuestro primer Notebook y podemos empezar a desarrollar código.



3. Instalación y primer contacto con el entorno de trabajo: Anaconda y Jupyter

Para establecer un nombre al notebook que estamos desarrollando debemos hacer click en Untitled100 y escribir el nombre deseado.



Cada celda puede ser de 4 tipos distintos:

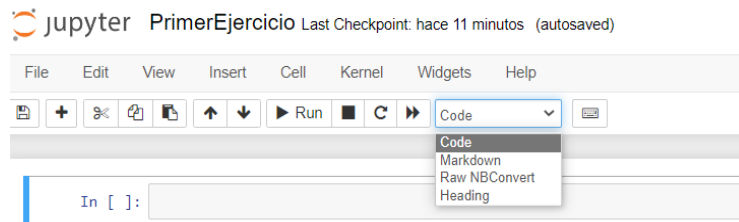
Code: sirve para escribir código ejecutable

Markdown: sirve para escribir texto

Raw NBConvert: escribe código que no se ejecutara

Heading: se utiliza para escribir titulos

Se seleccionan desde el menú desplegable:

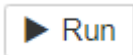


3. Instalación y primer contacto con el entorno de trabajo: Anaconda y Jupyter

Para crear nuevas celdas se utiliza el símbolo +



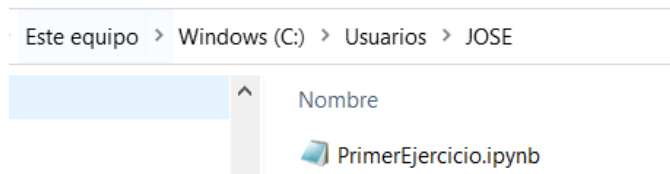
Para ejecutar la celda se utiliza el botón Run



Existen comandos especiales precedidos de % como %pwd que nos permite ejecutar comandos típicos de Linux. Este en este caso nos muestra el directorio de trabajo actual.

```
%pwd  
'C:\\Users\\JOSE'
```

Podemos ver que efectivamente se ha guardado aquí nuestro notebook.



...ahora...comencemos!

Instalemos Anaconda!

Abramos Jupyter y creemos un proyecto llamado MiPrimerPrograma!

