

Guía práctica: Primeros pasos con Python

Elaborado por: Lina María Castro

Asignatura: Inteligencia Artificial con Aplicaciones en Economía I
Universidad Externado de Colombia

Fecha: 5 de agosto de 2025

Objetivos de aprendizaje

- Instalar Python, VS Code y Git.
- Crear y activar un ambiente virtual.
- Conocer Jupyter Notebook e instalar librerías.
- Iniciar un proyecto con control de versiones en Git y publicarlo en GitHub.
- Clonar un repositorio de Github.
- Conocer Google Colab como alternativa para programar en la nube.

1. Introducción a Python y su utilidad en economía

Python es un lenguaje de programación versátil, gratuito y de código abierto. En economía, es ampliamente utilizado para análisis de datos, modelos econométricos, scraping de datos públicos y visualización de resultados. Algunas bibliotecas clave incluyen pandas, numpy, matplotlib, statsmodels y scikit-learn.

2. Instalación del entorno de trabajo

Sigue estos pasos para preparar tu entorno de desarrollo:

a. Instalar Python

- Ir a <https://www.python.org/downloads/>
- Descargar la última versión estable para el sistema operativo del computador.
- Abrir el ejecutable.
- Marcar “Add Python to PATH” e “Install Now”.
- Ir a cmd y escribir Python para verificar que haya quedado instalado.

b. Instalar Visual Studio Code

- Ir a <https://code.visualstudio.com/download>
- Descargar según sistema operativo del computador.
- Abrir el ejecutable e instalar.
- Abrir el programa para verificar que haya quedado instalado.

c. Instalar Git

- Ir a <https://git-scm.com/downloads>
- Descargar según sistema operativo del computador.
- Abrir el ejecutable e instalar.
- Abrir una cuenta en Github: <https://github.com/>
- Abrir cmd.
- Verificar que tenga instalado Git y cuál es la versión: `git version`
- Cambiar o agregar el nombre de usuario: `git config --global user.name "LinaMariaCastro"`
- Utilizar el correo asociado a Github: `git config --global user.email "lmcastroco@gmail.com"`
- Verificar que se haya registrado correctamente el usuario: `git config user.name`
- Verificar que se haya registrado correctamente el email: `git config user.email`
- Configurar la rama main como rama principal en lugar de master: `git config --global init.defaultBranch main`

3. Crear y activar un ambiente virtual

- Crear carpeta para tu proyecto/clase.
- Abrir VS Code, ir a Open Folder y ubicarse en la carpeta.
- Ir a File, New file, Jupyter Notebook.
- Ir a Select Kernel, Python environment, create Python environment, venv.
- Seleccionar la version de Python deseada (anotar cuál se usó).
- Click en "show logs" para ver el progreso de la creación del ambiente virtual. Cuando se detenga ya quedó listo (al final dice "commandPrompt").
- Cuando se abre un Jupyter Notebook, seleccionar en el Kernel el ambiente virtual creado.

4. Jupyter Notebook y uso de librerías

- Abrir un Jupyter Notebook.
- Instalar paquetes necesarios (ejemplo: `%pip install pandas`).
- Abrir una nueva terminal.
- `.venv\Scripts\activate` (en caso de que no esté activo)
- `python --version`
- Para ver todo lo instalado en el ambiente virtual: `pip freeze`

- Para crear el archivo de requerimientos de un proyecto con todos los programas necesarios a instalar: `pip freeze > requirements.txt`
- Cerrar terminal.
- En un nuevo notebook, importar librerías.
- Escribir y ejecutar el código deseado.

5. Crear un repositorio en GitHub

- En Visual Studio Code, ubicarme en la carpeta del proyecto y abrir una nueva terminal
- Crear un nuevo repositorio (crea la carpeta oculta `.git`, solo se ejecuta una vez por proyecto): `git init`
- Añadir todos los archivos de la carpeta al repositorio: `git add .`
- Nos dice todo lo que se añadió al repositorio, permite ver que archivos no han sido registrados: `git status -s`
- Crear una fotografía del proyecto en ese momento y colocarle un nombre de lo que contiene: `git commit -m "inicio del repositorio"`
- Ir a mi cuenta en Github y en el + escojo "create new repository", le doy un nombre al repositorio y le doy crear. No debo darle click en crear README, ese se puede crear después.
- En el repositorio en Github, copiar el https que aparece ahí.
- En la terminal de VS Code, indicar la ruta del repositorio en Github: `git remote add origin https://github.com/LinaMariaCastro/2025-segundo-semester-ia-economia.git`
- Indicar la rama donde va a quedar la información: `git branch -M main`
- Enviar la información al repositorio en Github: `git push -u origin main`
- Sale un cuadro de diálogo donde debo colocar las credenciales de mi cuenta en Github.

6. Subir cambios en el código local a Github

- `git add .`
- `git commit -m "agregue x cambio"`
- `git push`

7. Descargar cambios en el código en GitHub al equipo local

- `git pull origin main`

8. Clonar un repositorio existente

Clone: Descarga un repositorio en tu computador para trabajar localmente y sincronizar cambios con Github.

- Crear una carpeta en el computador donde voy a ubicar el proyecto.
- En Visual Studio Code, ubicarme en la carpeta del proyecto.
- Ir al repositorio en Github, hacer click en el botón verde "Code" y copiar el https que aparece ahí.
- Abrir una nueva terminal en VS Code.
- Escribir: git clone <https://github.com/LinaMariaCastro/prueba-clonar.git>
- Crear un ambiente virtual e instalar requirements.txt

9. Introducción a Google Colab

- Abrir <https://colab.research.google.com> y crear/subir un notebook.
- Ejecutar el código.
- Guardar el notebook en Google Drive o descargar.