



Pontificia Universidad  
**JAVERIANA**  
Bogotá

Educación **Continua**  
Generamos experiencias educativas

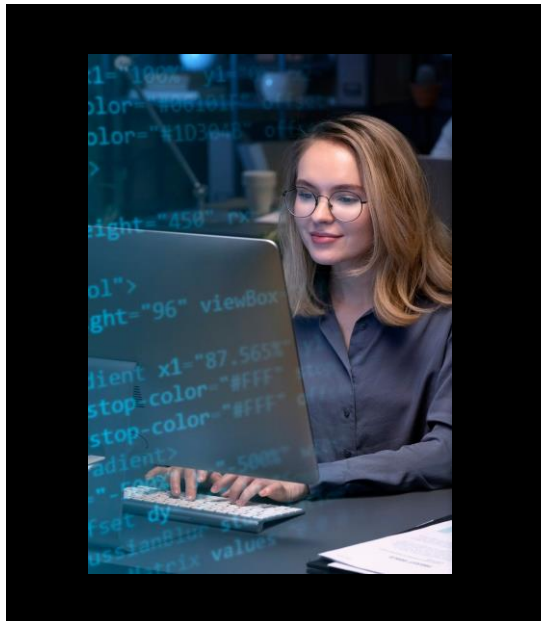
# Introducción a Python y Google Colab

Comunicaciones

# Introducción

En esta presentación se abordarán los fundamentos de Python y Google Colab, así como sus aplicaciones en la actualidad.

Se explorarán las ventajas de utilizar Google Colab para el desarrollo de proyectos en Python. ¡Bienvenidos!



# Python: Fundamentos

Python es un lenguaje de programación de alto nivel, interpretado y de propósito general.

## **Características principales:**

- Sencillo y legible.
- Interpretado y dinámico.
- Multiparadigma (soporta programación orientada a objetos, funcional y procedimental).
- Amplia biblioteca estándar.

Guido van Rossum

Comienza el  
proyecto de Python

Finales de  
los 80

Python 1.0

1994

Python 3.0

2008

Python 3.9

2021

1991

Primer  
Lanzamiento

2000

Python 2.0

2020

Fin de soporte  
Python 2.0

# Breve historia Python

# ¿Qué es un Entorno de Desarrollo?

Un entorno de desarrollo es un conjunto de herramientas que facilitan la escritura, ejecución y depuración de código de programación.

## Elementos de un IDE:

- Editor de código.
- Depurador.
- Compilador / intérprete.
- Integración con sistemas de control de versiones.
- Herramientas de pruebas.



Google Colab es una plataforma de código abierto que permite ejecutar y compartir scripts de Python.

**Características:**

- Basado en Jupyter Notebooks.
- Soporte para bibliotecas de machine learning y ciencia de datos.
- Integración con Google Drive.
- Acceso gratuito a GPUs y TPUs.

Visita [colab.research.google.com](https://colab.research.google.com)



# Python en local vs Google Colab

## Configuración:

- Requiere la instalación de Python y la configuración de entornos virtuales en tu máquina.
- Posibilidad de usar diversos IDEs locales como PyCharm, VS Code, Jupyter, Spyder, entre otros.



## Configuración:

- Servicio en la nube de Google basado en Jupyter Notebooks.
- No requiere instalación de software; accesible desde cualquier navegador con una cuenta de Google.





# Python en local vs Google Colab

## Ventajas:

- Personalización Completa: Control total sobre el entorno y configuraciones.
- Sin Dependencia de Internet: Puedes trabajar offline.
- Privacidad: Mayor control sobre la privacidad de tus datos y código
- Acceso a Herramientas Avanzadas: Posibilidad de integrar y usar herramientas específicas que no están disponibles en la nube.

## Ventajas:

- Acceso Universal: Trabaja desde cualquier lugar con conexión a internet.
- Recursos Avanzados Gratuitos: Acceso a GPU y TPU sin costo adicional, ideal para tareas intensivas en recursos.
- Facilidad de Colaboración: Comparte notebooks y colabora en tiempo real con otros usuarios.
- Integración con Google Drive: Guarda y carga archivos fácilmente desde tu Google Drive.





# Python en local vs Google Colab



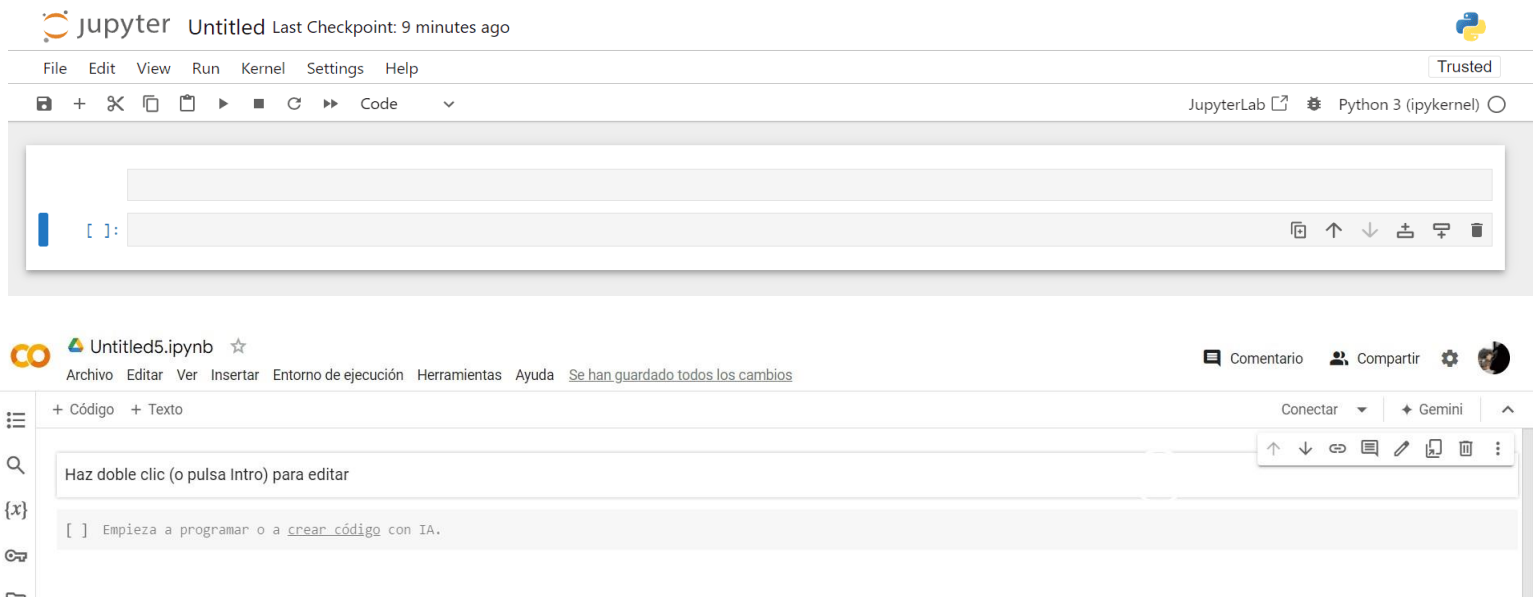
## Desventajas:

- Configuración Inicial: Requiere tiempo y esfuerzo para configurar el entorno.
- Requisitos de Hardware: Necesitas una máquina con suficiente capacidad de procesamiento y memoria.
- Mantenimiento: Debes gestionar actualizaciones y dependencias del sistema y bibliotecas.

## Desventajas:

- Dependencia de Internet: Necesitas estar conectado a internet para trabajar.
- Límites de Uso: Restricciones en el tiempo de uso y recursos disponibles en la versión gratuita.
- Menor Control y Personalización: Limitaciones en la personalización del entorno y configuración en comparación con entornos locales.

# Python en local vs Google Colab



The image displays two web-based development environments side-by-side. The top interface is JupyterLab, showing a menu bar with 'File', 'Edit', 'View', 'Run', 'Kernel', 'Settings', and 'Help'. Below the menu is a toolbar with icons for file operations and execution. The main area is a code editor with a Python prompt and a toolbar on the right. The bottom interface is Google Colab, showing a menu bar with 'Archivo', 'Editar', 'Ver', 'Insertar', 'Entorno de ejecución', 'Herramientas', 'Ayuda', and a link to 'Se han guardado todos los cambios'. Below the menu is a toolbar with icons for file operations and execution. The main area is a code editor with a text prompt and a toolbar on the right.

**JupyterLab** (top):

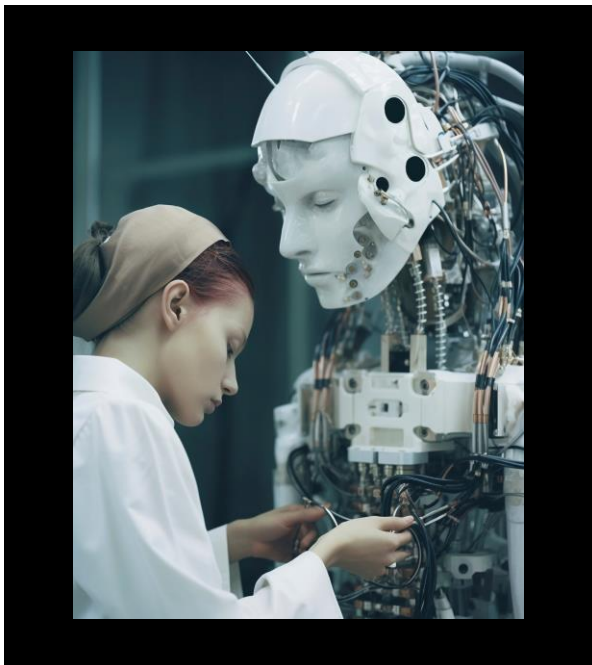
- Menu: File, Edit, View, Run, Kernel, Settings, Help
- Toolbar: File, Add, Copy, Paste, Run, Stop, Refresh, Code
- Status: JupyterLab Python 3 (ipykernel)
- Code Editor: [ ]:

**Google Colab** (bottom):

- Menu: Archivo, Editar, Ver, Insertar, Entorno de ejecución, Herramientas, Ayuda, [Se han guardado todos los cambios](#)
- Toolbar: Connect, Gemini, Up, Down, Link, Comment, Edit, Copy, Paste, More
- Code Editor: Haz doble clic (o pulsa Intro) para editar  
[ ] Empieza a programar o a [crear código](#) con IA.

### **Google Colab: Aplicaciones**

Google Colab es ideal para el aprendizaje de machine learning y la experimentación con modelos de inteligencia artificial. También es útil para el desarrollo de proyectos de análisis de datos y visualización. Su facilidad de uso lo hace accesible para principiantes.





## Consejos para Principiantes

Para aquellos que están comenzando, se recomienda explorar tutoriales en línea, participar en comunidades de programación y practicar regularmente. La comunidad de Python es acogedora y ofrece una gran cantidad de recursos.

# Conclusión

En resumen, Python y Google Colab son fundamentales para cualquier profesional que busque incursionar en el mundo de la programación y el análisis de datos. Con su versatilidad y accesibilidad, ofrecen un sinfín de posibilidades para el desarrollo de proyectos innovadores.