

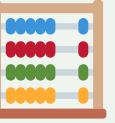
# INTRO A PYTHON PARA CIENCIA DE DATOS



# Clase 4:

## 11/05/2024



- Gestor de Paquetes
- Librerías Externas: Numpy 
- Librerías Externas: Pandas 
- Visualizaciones



# GESTOR DE PAQUETES



**LevelUP**  
Tech Academy

# GESTOR DE PAQUETES



Un gestor de paquetes en Python es una herramienta que facilita la instalación, actualización y eliminación de librerías y paquetes de Python.

Los paquetes son ***colecciones de módulos*** de Python que se pueden usar para añadir funcionalidades específicas a tus programas, sin necesidad de escribir todo el código desde cero.



# ¿POR QUÉ ES IMPORTANTE USAR UN GESTOR DE PAQUETES?



**Facilita el acceso a bibliotecas externas:** Puedes utilizar código desarrollado por otros para añadir nuevas funcionalidades a tus proyectos.

**Automatiza el proceso de instalación:** Con un solo comando, puedes instalar todas las dependencias necesarias para tu proyecto.

# ¿POR QUÉ ES IMPORTANTE USAR UN GESTOR DE PAQUETES?



**Mantiene las versiones organizadas:** Puedes gestionar diferentes versiones de paquetes para asegurarte de que tu proyecto funcione correctamente con las versiones correctas.

**Simplifica la colaboración:** Al usar un gestor de paquetes, puedes compartir tus proyectos con otros de manera más eficiente, ya que pueden replicar el entorno exacto que necesitas.

# COMANDOS BÁSICOS DE `PIP`



## Instalar un paquete

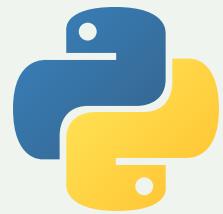
Para instalar un paquete, se usa el comando **pip install** seguido del nombre del paquete.

Por ejemplo, para instalar la librería pandas, se escribiría:

**`pip install pandas`**



# COMANDOS BÁSICOS DE `PIP`



## - Actualizar un paquete

Para actualizar un paquete a la última versión disponible, se usa el comando pip install --upgrade seguido del nombre del paquete. Por ejemplo:

**`pip install --upgrade pandas`**



# COMANDOS BÁSICOS DE `PIP`



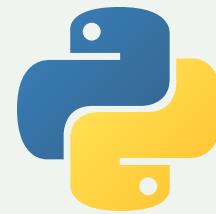
## Desinstalar un paquete

Para desinstalar un paquete, se usa el comando **pip uninstall** seguido del nombre del paquete. Por ejemplo:

**`pip uninstall pandas`**



# COMANDOS BÁSICOS DE `PIP`



## Listar los paquetes instalados

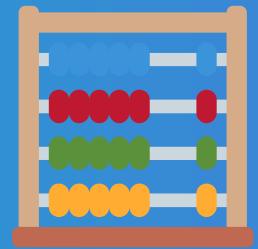
Para ver una lista de todos los paquetes instalados en tu entorno de Python, se usa el comando pip list:

**`pip list`**

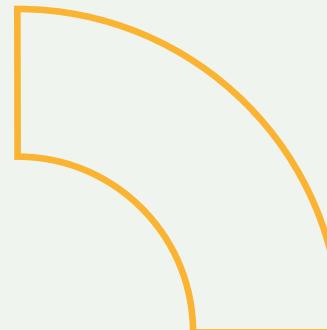


Package	Version
absl-py	1.4.0
aiohttp	3.8.4
aiohttp-retry	2.8.3
aiosignal	1.3.1
anyio	3.7.0
argon2-cffi	23.1.0
argon2-cffi-bindings	21.2.0
arrow	1.2.3
astor	0.8.1
asttokens	2.2.1
astunparse	1.6.3
async-generator	1.10

# LIBRERÍAS EXTERNAS: NUMPY



**LevelUP**  
Tech Academy



# NUMPY



NumPy es una ***librería fundamental*** para la computación científica en Python.

Su nombre proviene de "**Numerical Python**" y proporciona soporte para **arrays multidimensionales** (también conocidos como matrices) y una variedad de funciones matemáticas y estadísticas que se pueden aplicar a estos arrays.

# ¿PARA QUÉ SE USA NUMPY?

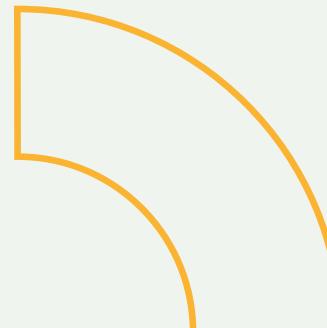


- **Manipulación de Arrays:** Ofrece un objeto array n-dimensional, que es mucho más eficiente y flexible que las listas de Python para manejar datos numéricos.
- **Operaciones Matemáticas y Estadísticas:** Permite realizar operaciones matemáticas y estadísticas complejas de manera eficiente.
- **Integración con Otras Librerías:** NumPy es la base sobre la cual se construyen muchas otras librerías de Python usadas en ciencia de datos, como **Pandas**, **Matplotlib** y **Scikit-learn**.

# LIBRERÍAS EXTERNAS: PANDAS



**LevelUP**  
Tech Academy



# PANDAS



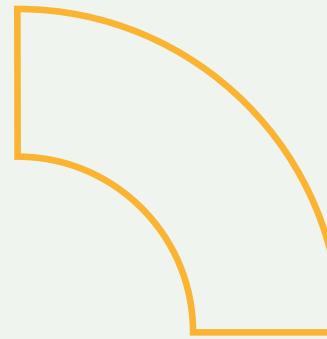
Pandas es una librería de Python utilizada para la manipulación y el ***análisis de datos***.

Su nombre proviene de "**Panel Data**" (datos en panel), y está diseñada para trabajar de manera eficiente con datos estructurados, como tablas y series temporales.

# ¿PARA QUÉ SE USA PANDAS?



- **Manipulación y Análisis de Datos:** Facilita la limpieza, transformación y análisis de datos.
- **Estructuras de Datos Flexibles:** Proporciona estructuras de datos eficientes, como **DataFrames** y **Series**.
- **Lectura y Escritura de Datos:** Permite importar y exportar datos desde y hacia diversos formatos, como CSV, Excel, SQL, JSON, entre otros.



# VISUALIZACIONES



**LevelUP**  
Tech Academy

# VISUALIZACIONES



Con la librería pandas, podemos crear directamente visualizaciones que nos van a permitir explorar nuestros datos, analizarlos y determinar hipótesis.

Además existen otras librerías utilizadas para la parte de visualización, en las que no entraremos en éste curso como ***matplotlib*** y ***sklearn***





Team Level Up

LevelUP  
Tech Academy