



MATERIAL DE LA CLASE

INTELIGENCIA ARTIFICIAL

CLASE 4

Bienvenidos a la inmersión de la Semana de Python en la Práctica

¡Es un placer tenerte aquí con nosotros en esta inmersión!

Este folleto pretende presentar de forma sencilla y directa todo el contenido que se vio en la clase en vivo.

Aquí en Daxus Latam creemos firmemente en el aprendizaje basado en proyectos y por eso, cada día de inmersión trae un nuevo proyecto a desarrollar.

Recuerde siempre en nuestra metodología ORA:

OBSERVAR: Durante la clase en vivo, simplemente siga lo que se está haciendo y tome nota de las explicaciones.

REPETIR: luego, rehaz el proyecto con el material en mano y si tienes alguna dificultad, repasa la clase. Todas las clases se graban y están disponibles en YouTube durante la semana de inmersión.

APLICAR: Ahora que has observado y repetido, es hora de aplicar lo aprendido. Crear nuevos proyectos con los conocimientos adquiridos.

¿Lo que vamos a aprender?

- › Usar las bibliotecas OpenCV, MediaPipe y CVzone
- › Utilizar la cámara web de la computadora con Python
- › Crear un código de visión computacional
- › Hacer el seguimiento de manos en tiempo real

Visión Computacional

La Visión Computacional es un campo de la Inteligencia Artificial enfocado en capacitar a las máquinas para interpretar y entender el mundo a través de imágenes y videos.

Python es uno de los mejores lenguajes para la Visión Computacional debido a sus bibliotecas robustas y fáciles de usar, como OpenCV, MediaPipe y CVzone, que ofrecen herramientas poderosas para el procesamiento de imágenes, reconocimiento de patrones y desarrollo de modelos de aprendizaje profundo.

Bibliotecas que utilizaremos

Instalando las bibliotecas Pandas, OpenpyXL y Plotly Express

TERMINAL

```
pip install opencv-python
```

```
pip install mediapipe
```

```
pip install cvzone
```

Corrección del problema de compatibilidad de la versión 3.14

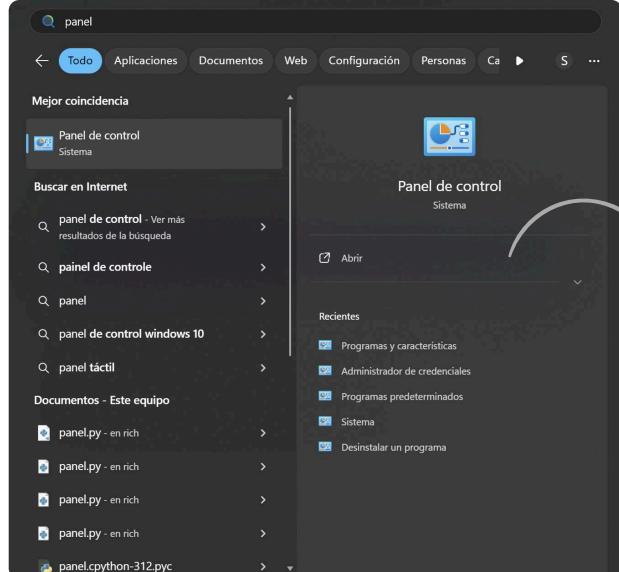
Por qué es necesario hacer esta instalación:

Si instalaron Python recientemente y cuentan **3.12.x, 3.13.x o 3.14.x** se genera un error:

```
C:\Users\pczak\Desktop\prueba>py proyecto04.py
Traceback (most recent call last):
  File "C:\Users\pczak\Desktop\prueba\proyecto04.py", line 12, in <module>
    rastreador = HandDetector(detectionCon=0.8, maxHands=2)
  File "C:\Users\pczak\AppData\Local\Programs\Python\Python313\Lib\site-packages\cvzone\HandTrackingModule.py", line 35, in __init__
    self.mpHands = mp.solutions.hands
                           ^^^^^^^^^^
AttributeError: module 'mediapipe' has no attribute 'solutions'
```

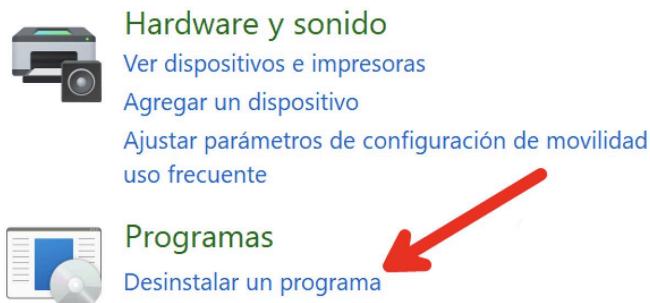
Este error ocurre por una incompatibilidad de arquitectura en la versión de Python y la librería. Para solucionar este error debemos desinstalar la versión de Python que tenemos actualmente y volver a instalar otra versión.

Proceso de desinstalación de Python e instalación:

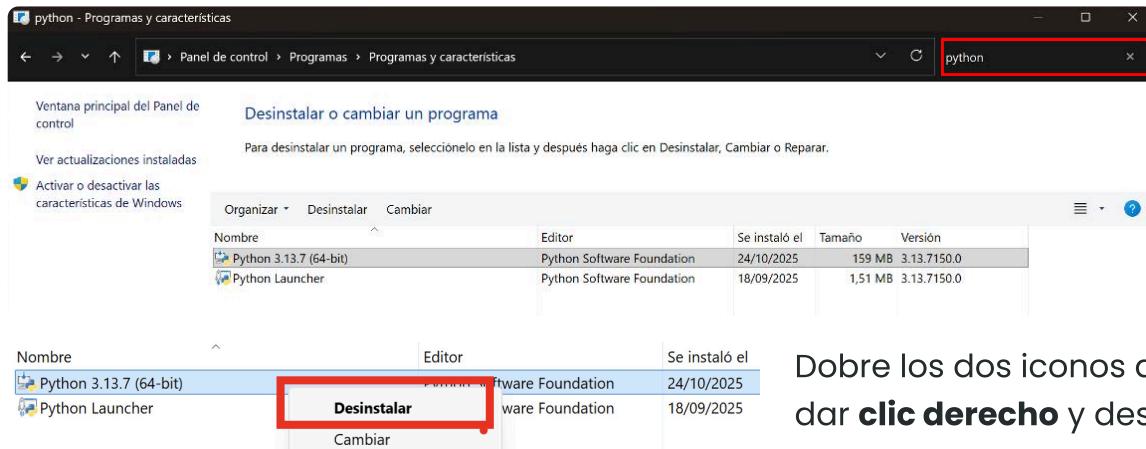


Buscas en tu barra de navegación en el SO **“Panel de control”**.

Luego te diriges a programas y presionas **“Desinstalar un programa”**:

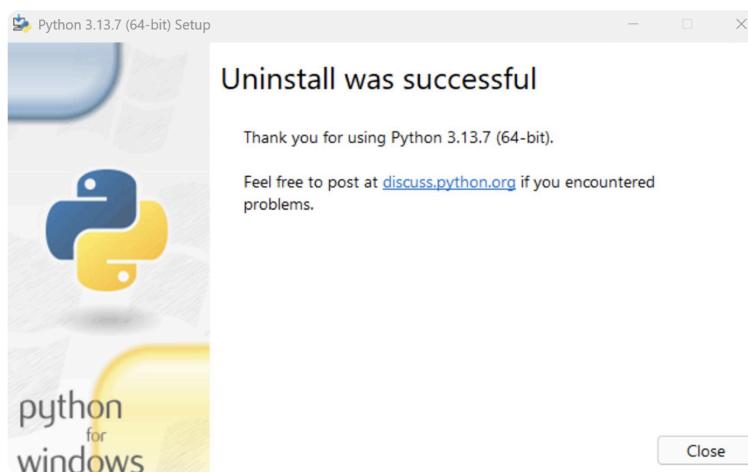


En el buscador de la parte superior derecha escriben “Python”:

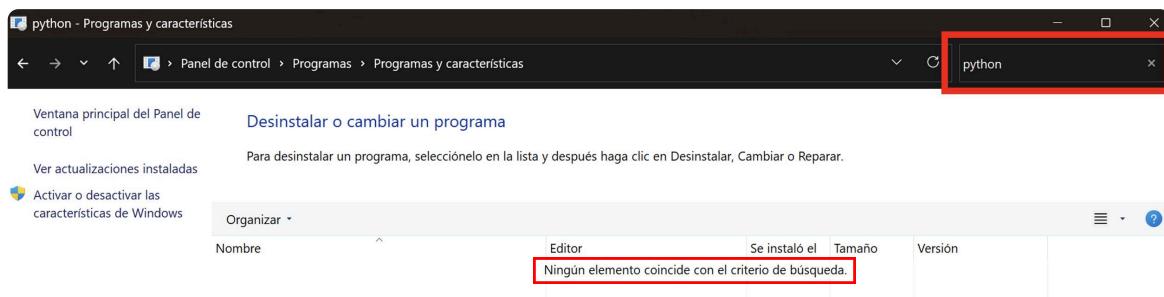


Dobre los dos iconos de “Python” dar **clic derecho** y desinstalar.

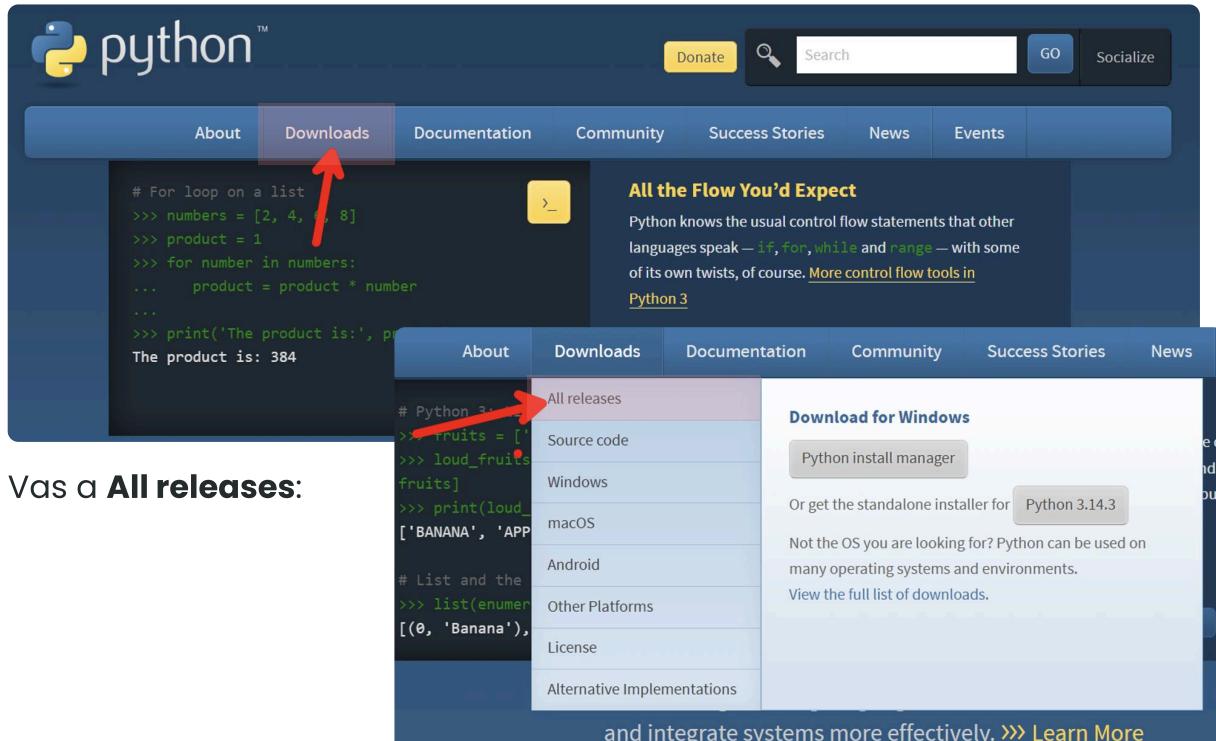
Se abrirá una pestaña donde se va a desinstalar esta versión, esperan a que cargue y aparezca el mensaje de “*Uninstall was successful*”:



Lo mismo con el launcher y su panel de control **debe quedar vacío** para Python:



Ya abres la página de Python (<https://www.python.org/>) vas a Downloads:



Vas a All releases:

All the Flow You'd Expect

Python knows the usual control flow statements that other languages speak — `if`, `for`, `while` and `range` — with some of its own twists, of course. [More control flow tools in Python 3](#)

About **Downloads** Documentation Community Success Stories News Events

For loop on a list
 >>> numbers = [2, 4, 6, 8]
 >>> product = 1
 >>> for number in numbers:
 ... product = product * number
 ...
 >>> print('The product is:', product)
 The product is: 384

Python 3.14.3
 >>> fruits = ['apple', 'banana', 'orange', 'grape']
 >>> loud_fruits = [fruit.upper() for fruit in fruits]
 >>> print(loud_fruits)
 ['APPLE', 'BANANA', 'ORANGE', 'GRAPE']
 # List and the range function
 >>> list(enumerate([(0, 'Banana'), (1, 'Apple'), (2, 'Orange'), (3, 'Grape')]))
 [(0, 'Banana'), (1, 'Apple'), (2, 'Orange'), (3, 'Grape')]

All releases

Source code

Windows

macOS

Android

Other Platforms

License

Alternative Implementations

Download for Windows

Python install manager

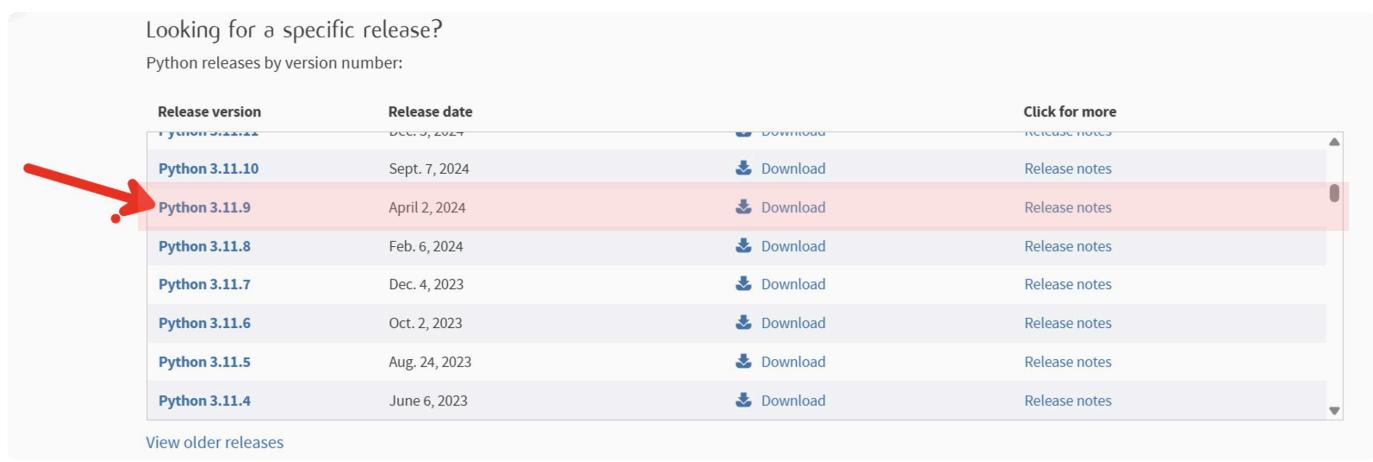
Or get the standalone installer for Python 3.14.3

Not the OS you are looking for? Python can be used on many operating systems and environments.

[View the full list of downloads.](#)

and integrate systems more effectively. [Learn More](#)

Bajas hasta encontrar “**Looking for a specific release?**” y selesccionas la versión **3.11.9** o cualquiera anterior a esa.



Looking for a specific release?

Python releases by version number:

Release version	Release date	Click for more
Python 3.14.3	Dec. 5, 2024	Release notes
Python 3.11.10	Sept. 7, 2024	Download Release notes
Python 3.11.9	April 2, 2024	Download Release notes
Python 3.11.8	Feb. 6, 2024	Download Release notes
Python 3.11.7	Dec. 4, 2023	Download Release notes
Python 3.11.6	Oct. 2, 2023	Download Release notes
Python 3.11.5	Aug. 24, 2023	Download Release notes
Python 3.11.4	June 6, 2023	Download Release notes

[View older releases](#)

Una vez ahí, bajas hasta **files** y buscas el **instalador de Windows 64-bits** o el que corresponda en tu caso:

Files

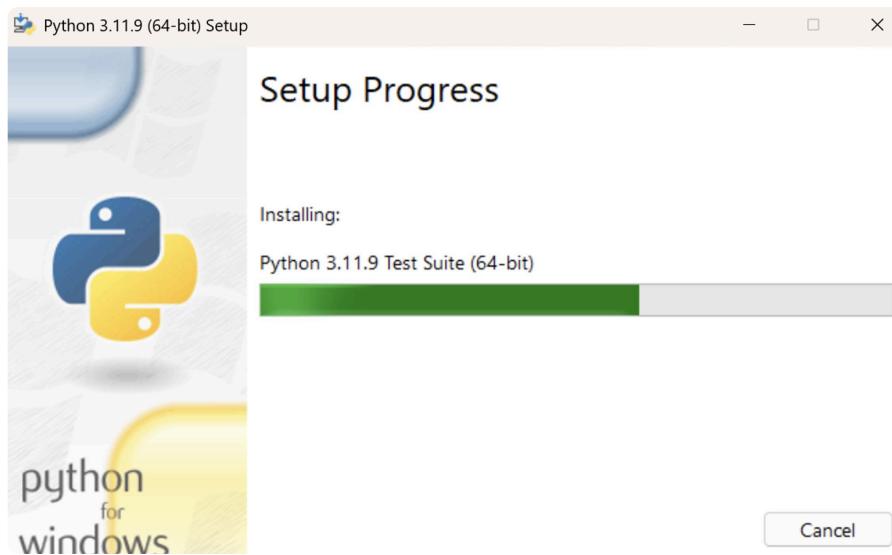
macOS	Windows	Source release																																																																						
Download macOS 64-bit universal2 installer	Download Python install manager	Download XZ compressed source tarball																																																																						
<table border="1"> <thead> <tr> <th>Version</th> <th>Operating system</th> <th>Description</th> <th>File size</th> <th>Sigstore</th> <th>GPG</th> <th>MD5 checksum</th> </tr> </thead> <tbody> <tr> <td>Gzipped source tarball</td> <td>Source release</td> <td></td> <td>25.3 MB</td> <td>.sigstore</td> <td>SIG</td> <td>bfd4d3bfeac4216ce35d7a503bf02d5c</td> </tr> <tr> <td>XZ compressed source tarball</td> <td>Source release</td> <td></td> <td>19.2 MB</td> <td>.sigstore</td> <td>SIG</td> <td>22ea467e7d915477152e99d5da856ddc</td> </tr> <tr> <td>macOS 64-bit universal2 installer</td> <td>macOS</td> <td>for macOS 10.9 and later</td> <td>42.8 MB</td> <td>.sigstore</td> <td>SIG</td> <td>fa29f456feb6b5c4f52456a8b8ba347b</td> </tr> <tr style="background-color: #fce4ec;"> <td>Windows installer (64-bit)</td> <td>Windows</td> <td>Recommended</td> <td>25.0 MB</td> <td>.sigstore</td> <td>SIG</td> <td>e8dc5d502e34932eebcfa1be056d5cbcd</td> </tr> <tr> <td>Windows installer (32-bit)</td> <td>Windows</td> <td></td> <td>23.8 MB</td> <td>.sigstore</td> <td>SIG</td> <td>2a1d1ac2d8a0aa847515f9dd121ccb7</td> </tr> <tr> <td>Windows installer (ARM64)</td> <td>Windows</td> <td>Experimental</td> <td>24.3 MB</td> <td>.sigstore</td> <td>SIG</td> <td>328d93f71cb078965e4cfa2eb2663fa1</td> </tr> <tr> <td>Windows embeddable package (64-bit)</td> <td>Windows</td> <td></td> <td>10.7 MB</td> <td>.sigstore</td> <td>SIG</td> <td>6d9aa08531d48fcc261ba667e2df17c4</td> </tr> <tr> <td>Windows embeddable package (32-bit)</td> <td>Windows</td> <td></td> <td>9.6 MB</td> <td>.sigstore</td> <td>SIG</td> <td>31e7648158376e92a4463aa6f22a78e1</td> </tr> <tr> <td>Windows embeddable package (ARM64)</td> <td>Windows</td> <td></td> <td>10.0 MB</td> <td>.sigstore</td> <td>SIG</td> <td>8611b6aa35483ab1c61d45e0d9f2de0d</td> </tr> </tbody> </table>			Version	Operating system	Description	File size	Sigstore	GPG	MD5 checksum	Gzipped source tarball	Source release		25.3 MB	.sigstore	SIG	bfd4d3bfeac4216ce35d7a503bf02d5c	XZ compressed source tarball	Source release		19.2 MB	.sigstore	SIG	22ea467e7d915477152e99d5da856ddc	macOS 64-bit universal2 installer	macOS	for macOS 10.9 and later	42.8 MB	.sigstore	SIG	fa29f456feb6b5c4f52456a8b8ba347b	Windows installer (64-bit)	Windows	Recommended	25.0 MB	.sigstore	SIG	e8dc5d502e34932eebcfa1be056d5cbcd	Windows installer (32-bit)	Windows		23.8 MB	.sigstore	SIG	2a1d1ac2d8a0aa847515f9dd121ccb7	Windows installer (ARM64)	Windows	Experimental	24.3 MB	.sigstore	SIG	328d93f71cb078965e4cfa2eb2663fa1	Windows embeddable package (64-bit)	Windows		10.7 MB	.sigstore	SIG	6d9aa08531d48fcc261ba667e2df17c4	Windows embeddable package (32-bit)	Windows		9.6 MB	.sigstore	SIG	31e7648158376e92a4463aa6f22a78e1	Windows embeddable package (ARM64)	Windows		10.0 MB	.sigstore	SIG	8611b6aa35483ab1c61d45e0d9f2de0d
Version	Operating system	Description	File size	Sigstore	GPG	MD5 checksum																																																																		
Gzipped source tarball	Source release		25.3 MB	.sigstore	SIG	bfd4d3bfeac4216ce35d7a503bf02d5c																																																																		
XZ compressed source tarball	Source release		19.2 MB	.sigstore	SIG	22ea467e7d915477152e99d5da856ddc																																																																		
macOS 64-bit universal2 installer	macOS	for macOS 10.9 and later	42.8 MB	.sigstore	SIG	fa29f456feb6b5c4f52456a8b8ba347b																																																																		
Windows installer (64-bit)	Windows	Recommended	25.0 MB	.sigstore	SIG	e8dc5d502e34932eebcfa1be056d5cbcd																																																																		
Windows installer (32-bit)	Windows		23.8 MB	.sigstore	SIG	2a1d1ac2d8a0aa847515f9dd121ccb7																																																																		
Windows installer (ARM64)	Windows	Experimental	24.3 MB	.sigstore	SIG	328d93f71cb078965e4cfa2eb2663fa1																																																																		
Windows embeddable package (64-bit)	Windows		10.7 MB	.sigstore	SIG	6d9aa08531d48fcc261ba667e2df17c4																																																																		
Windows embeddable package (32-bit)	Windows		9.6 MB	.sigstore	SIG	31e7648158376e92a4463aa6f22a78e1																																																																		
Windows embeddable package (ARM64)	Windows		10.0 MB	.sigstore	SIG	8611b6aa35483ab1c61d45e0d9f2de0d																																																																		

Buscas el instalador en tus descargas:

Lo abres y te sale el instalador de Python, recuerda selección **Add python.exe to PATH** y luego dar a instalar.

The screenshot shows the 'Python 3.11.9 (64-bit) Setup' window. At the top, it says 'Install Python 3.11.9 (64-bit)' and 'Select Install Now to install Python with default settings, or choose Customize to enable or disable features.' Below this, there are two main options: 'Install Now' (which has a red arrow pointing to it) and 'Customize installation'. Under 'Install Now', it says 'Includes IDLE, pip and documentation' and 'Creates shortcuts and file associations'. At the bottom, there are two checked checkboxes: 'Use admin privileges when installing py.exe' and 'Add python.exe to PATH' (which also has a red arrow pointing to it). A 'Cancel' button is at the bottom right.

Esperas a que se instale.



Luego **ejecutamos en la terminal:**

```
pip install opencv-python cvzone mediapipe==0.10.11
```

Y ahora ya no tendremos este problema para practicar el proyecto de la clase.

Bucles de repetición (loopings) con while

SALIDA

```
contador = 5
while contador > 0:
    print(contador)
    contador = contador + 1
```

SAÍDA

```
5
4
3
2
1
```

Pruebas condicionales con if

ENTRADA

```
lenguaje = "Python"
if lenguaje == "Python":
    print("¡Mejor lenguaje del mundo!")
```

SALIDA

¡Mejor lenguaje del mundo!

Bucles infinitos con while e if

ENTRADA

```
while True:
    texto = input("Escribe algo: ")
    print(texto)
    if texto == "parar":
        break
```

Imagina que en este ejemplo escribimos los siguientes textos

- › prueba
- › seguir
- › parar

SALIDA

prueba
seguir
parar

Es muy común usar bucles infinitos cuando estamos desarrollando soluciones de visión computacional que utilizan una cámara para capturar imágenes.

Esto ocurre porque necesitamos capturar continuamente cuadros para procesarlos en tiempo real.

Código del proyecto

```
import cv2
from cvzone.HandTrackingModule import HandDetector

# Inicializa la cámara web
webcam = cv2.VideoCapture(0)

# Inicializa el rastreador de manos
rastreador = HandDetector(detectionCon=0.8, maxHands=2)

while True:
    # captura la imagen de la cámara web
    éxito, imagen = webcam.read()

    # Detecta las manos en el cuadro
    hands, imagen_manos = rastreador.findHands(imagen)

    # Muestra el cuadro con las marcas
    cv2.imshow("Proyecto 4 - IA", imagen_manos)

    # Cierra la aplicación cuando se presiona cualquier tecla
    if cv2.waitKey(1) != -1:
        break

# Libera la cámara y cierra las ventanas
webcam.release()
cv2.destroyAllWindows()
```



**Una inmersión de 4 clases para que aprendas
Python desde cero y crees proyectos que
destaquen en el mercado laboral, incluso si
nunca has programado antes.**



TODOS LOS DERECHOS RESERVADOS **WWW.DAXUS.COM**. ESTE CONTENIDO ES **PROPIEDAD DE DAXUS LATAM** – SE PROHÍBE SU COMERCIALIZACIÓN Y REPRODUCCIÓN TOTAL O PARCIAL.