



Módulo 1 - Unidad 2 - Lección 1

Actividad 1

Introducción a Python

Contextualización sobre Python como un lenguaje versátil y de alto nivel

Python, conocido por su sintaxis clara y legible, se ha convertido en uno de los lenguajes de programación más populares y versátiles en la industria del software. Su diseño centrado en la legibilidad y su sintaxis concisa hacen que Python sea accesible incluso para aquellos que están dando sus primeros pasos en programación.



Versatilidad



Interpretación



Lenguajes
compilados



Lenguajes
interpretados



Lenguajes Compilados

Definición: los lenguajes compilados requieren un paso adicional antes de la ejecución. El código fuente escrito por el programador se traduce completamente a código de máquina u otro código intermedio por un programa llamado compilador.

Proceso: el programador escribe el código fuente en un archivo con extensión como ".c" (en C) o ".java" (en Java). El compilador toma ese código fuente y lo traduce a código de máquina o a otro lenguaje de bajo nivel, se genera un archivo ejecutable que puede ser ejecutado directamente por la máquina.

Ventajas: mayor eficiencia de ejecución, ya que el código se traduce completamente antes de la ejecución. El código compilado suele ser más rápido.

Ejemplos: C, C++, Java (en parte, ya que utiliza la compilación Just-In-Time - JIT), Fortran.



Interpretación

Python es un lenguaje interpretado, lo que significa que el código se ejecuta línea por línea, facilitando el desarrollo interactivo y la depuración. La clasificación de lenguajes de programación en "compilados e interpretados", se refiere al proceso mediante el cual el código fuente escrito por el programador se traduce y ejecuta.



Lenguajes Interpretados

Definición: los lenguajes interpretados no requieren un paso de compilación antes de la ejecución. En cambio, un programa llamado intérprete lee y ejecuta directamente el código fuente.

Proceso: el programador escribe el código fuente en un archivo con extensión como ".py" (en Python) o ".js" (en JavaScript). Un intérprete lee el código fuente línea por línea y lo ejecuta directamente sin generar un archivo ejecutable.

Ventajas: mayor portabilidad, ya que el código fuente puede ejecutarse en cualquier sistema que tenga el intérprete instalado y obtener facilidad de depuración, ya que se puede ejecutar el código línea por línea.

Ejemplos: Python, JavaScript, Ruby, PHP.



Versatilidad

Lo que distingue a Python es su versatilidad; puede ser utilizado en una variedad de contextos, desde desarrollo web y scripting hasta análisis de datos y aprendizaje automático. La capacidad de Python para adaptarse a diferentes dominios lo convierte en una elección natural para programadores, científicos de datos e ingenieros de software.

La clasificación de lenguajes de programación en compilados e interpretados se centra en si el código fuente se traduce completamente antes de la ejecución (compilados) o si se interpreta y ejecuta línea por línea (interpretados). Cada enfoque tiene sus propias ventajas y desventajas, y la elección entre ambos depende de los requisitos y objetivos del proyecto.

Python es un lenguaje de alto nivel, lo que implica que ofrece abstracciones poderosas que simplifican la tarea del programador. Esta característica permite a los desarrolladores concentrarse en la lógica del problema en lugar de preocuparse por detalles de bajo nivel, la clasificación de los lenguajes de programación en "niveles bajo y alto" se refiere a la proximidad del lenguaje al lenguaje máquina y al hardware de la computadora.



Nivel bajo



Nivel alto



Comunidad Activa y
Bibliotecas Abundantes





Lenguajes de Nivel Alto

Definición: los lenguajes de nivel alto están diseñados para ser más accesibles y abstraen al programador de los detalles de la arquitectura del hardware.

Características:

- Utilizan una sintaxis más cercana al lenguaje humano, facilitando la escritura y comprensión del código.
- Proporcionan abstracciones y funciones de más alto nivel, lo que permite a los programadores enfocarse en la lógica del problema más que en los detalles de implementación de la máquina.
- Son independientes de la arquitectura del hardware, lo que facilita la portabilidad del código entre diferentes plataformas.

Ejemplos: Python, Java, C++, y JavaScript son ejemplos de lenguajes de nivel alto. Estos lenguajes se centran en la productividad del programador y son más utilizados en el desarrollo de aplicaciones y sistemas de software.



Lenguajes de Nivel Bajo

Definición: los lenguajes de nivel bajo están más cerca del lenguaje de máquina y del hardware de la computadora.

Características: Utilizan una sintaxis y estructuras de programación que reflejan directamente las operaciones ejecutadas por la unidad central de procesamiento (CPU).

Ejemplos: ensamblador es un lenguaje de nivel bajo que representa instrucciones de máquina de manera simbólica. También se consideran de nivel bajo lenguajes como C, que, aunque más alto que el ensamblador, permite un control más directo sobre la memoria y las operaciones de la CPU.



Comunidad Activa y Bibliotecas Abundantes

Python ha construido una comunidad vibrante y colaborativa, esta comunidad activa contribuye a la mejora constante del lenguaje y proporciona una rica colección de bibliotecas y frameworks que facilitan el desarrollo; bibliotecas como NumPy para computación científica, Pandas para manipulación de datos y TensorFlow para aprendizaje automático, ilustran la amplitud y profundidad de las herramientas disponibles en Python.

Facilidad de Aprendizaje:

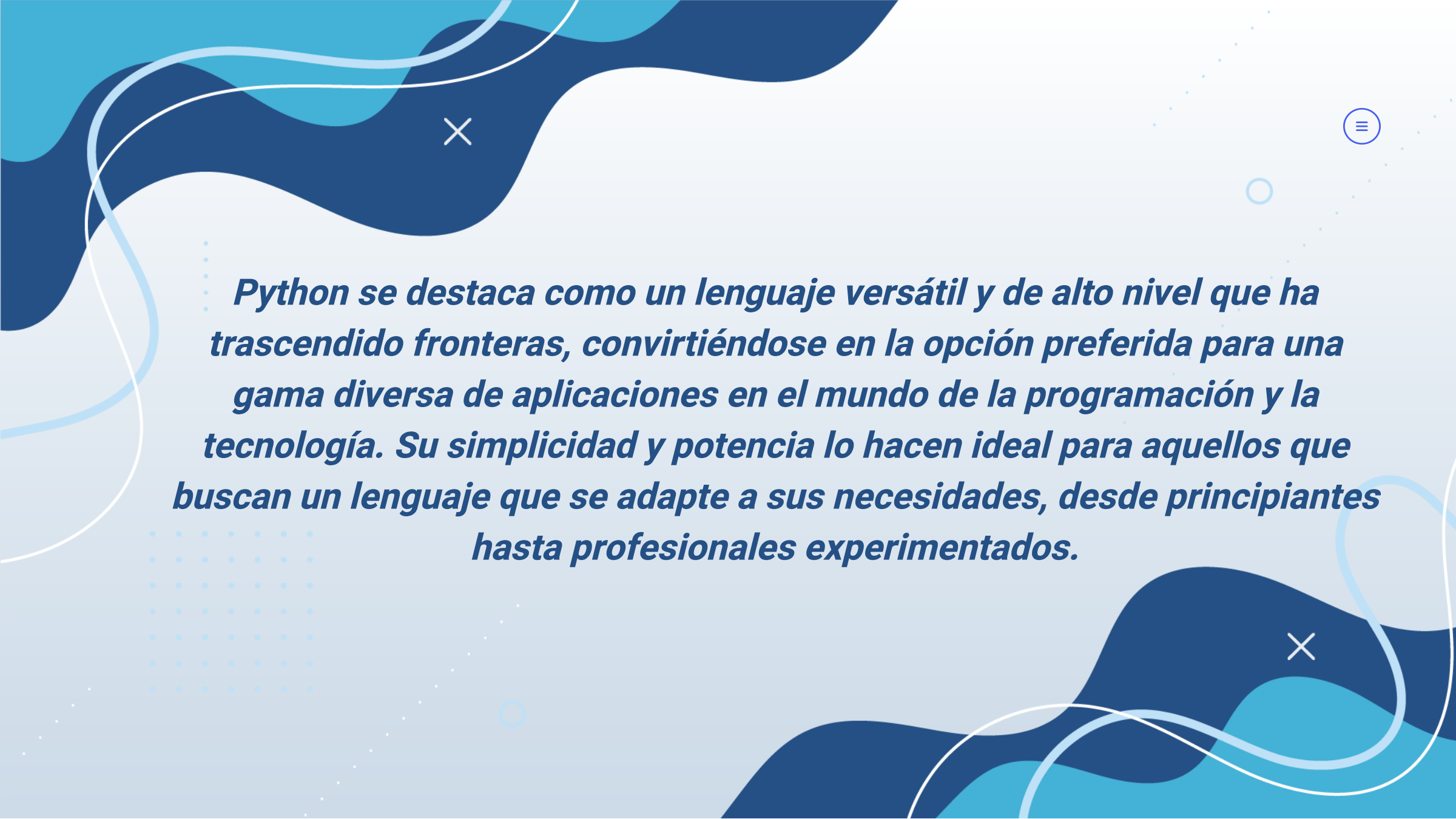
Python ha construido una comunidad vibrante y colaborativa, esta comunidad activa contribuye a la mejora constante del lenguaje y proporciona una rica colección de bibliotecas y frameworks que facilitan el desarrollo; bibliotecas como NumPy para computación científica, Pandas para manipulación de datos y TensorFlow para aprendizaje automático, ilustran la amplitud y profundidad de las herramientas disponibles en Python.

Adaptabilidad en la Industria:

La adopción generalizada de Python en la industria tecnológica es evidencia de su impacto. Empresas líderes utilizan Python para una variedad de aplicaciones, desde desarrollo web hasta inteligencia artificial. Su capacidad para integrarse con otros lenguajes y sistemas hace que sea una elección pragmática y poderosa para proyectos de software a gran escala.



La clasificación de lenguajes de programación en niveles bajo y alto se basan en la relación entre la sintaxis del lenguaje y la proximidad a las operaciones de la máquina, brindando a los programadores diferentes niveles de control y abstracción.



Python se destaca como un lenguaje versátil y de alto nivel que ha trascendido fronteras, convirtiéndose en la opción preferida para una gama diversa de aplicaciones en el mundo de la programación y la tecnología. Su simplicidad y potencia lo hacen ideal para aquellos que buscan un lenguaje que se adapte a sus necesidades, desde principiantes hasta profesionales experimentados.

Python es un Pilar Indiscutible en Inteligencia Artificial y Ciencia de Datos

En el fascinante universo de la inteligencia artificial y la ciencia de datos, Python se alza como un pilar indiscutible. Su presencia en estas disciplinas es omnipresente, alimentando innovaciones y descubrimientos que transforman nuestra comprensión del mundo.



Prototipado y
Experimentación



Bibliotecas
Potentes



Aprendizaje Automático
y Redes Neuronales



Comunidad
Activa



Empresas e
Investigación



Aprendizaje Automático y Redes Neuronales

En el ámbito del aprendizaje automático, Python es el lenguaje por excelencia. La flexibilidad de bibliotecas como Scikit-Learn permite implementar algoritmos supervisados y no supervisados con facilidad.

Además, la popularidad de TensorFlow y PyTorch ha llevado a un auge en el uso de Python para desarrollar y entrenar redes neuronales profundas, impulsando el avance del aprendizaje profundo.



Adaptabilidad a Empresas e Investigación

Python no solo lidera en empresas, sino que también es la elección preferida en entornos de investigación académica. Su versatilidad, la posibilidad de integrarse con otros lenguajes y sistemas, su capacidad para manejar grandes volúmenes de datos hacen que sea una opción pragmática y efectiva tanto para experimentos en laboratorios como para implementaciones a escala industrial.



Comunidad Activa

La comunidad Python en el ámbito de la inteligencia artificial y la ciencia de datos es extraordinariamente activa y colaborativa. Esto se traduce en una rápida evolución del lenguaje y la constante aparición de nuevas herramientas y técnicas, la participación activa de la comunidad garantiza que Python siga siendo relevante y poderoso en la resolución de los desafíos más complejos.



Bibliotecas Potentes

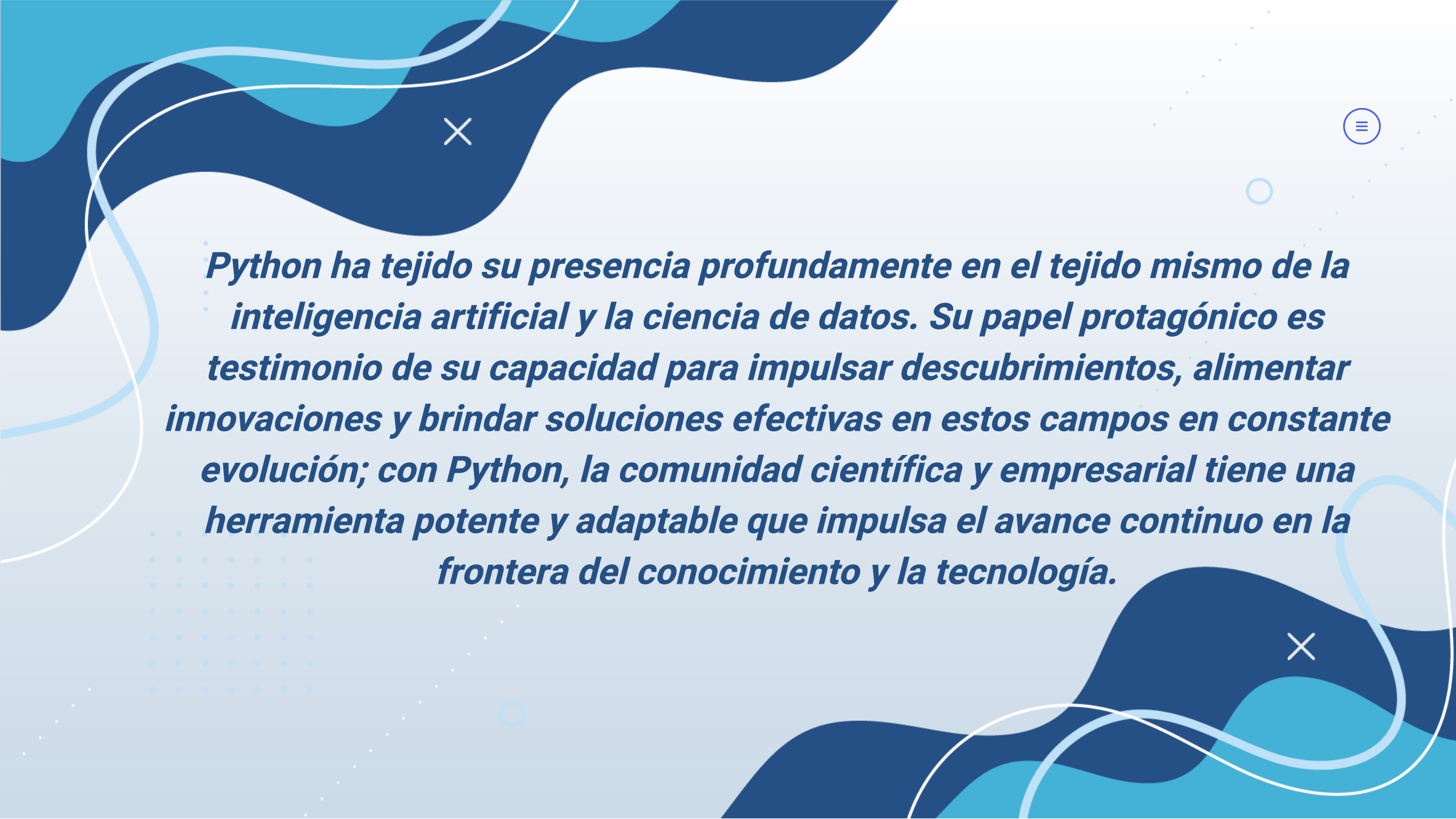
Lo que realmente distingue a Python en estos campos es su ecosistema de bibliotecas especializadas, NumPy, Pandas y Matplotlib forman el núcleo de manipulación de datos y visualización, SciPy proporciona herramientas para la ciencia y la ingeniería, mientras que Scikit-Learn ofrece algoritmos de aprendizaje automático listos para usar.

TensorFlow y PyTorch, dos frameworks líderes, están escritos en Python, consolidando aún más su posición en la vanguardia de la inteligencia artificial.



Facilidad de Prototipado y Experimentación

Python se ha ganado la reputación de ser un lenguaje ideal para la fase inicial de prototipado y experimentación. La sintaxis clara y la velocidad de desarrollo permiten a los científicos de datos e ingenieros de aprendizaje automático traducir rápidamente ideas en código funcional. La capacidad de experimentar sin obstáculos es clave para impulsar la innovación y descubrir soluciones efectivas.



Python ha tejido su presencia profundamente en el tejido mismo de la inteligencia artificial y la ciencia de datos. Su papel protagónico es testimonio de su capacidad para impulsar descubrimientos, alimentar innovaciones y brindar soluciones efectivas en estos campos en constante evolución; con Python, la comunidad científica y empresarial tiene una herramienta potente y adaptable que impulsa el avance continuo en la frontera del conocimiento y la tecnología.



TIC

TALENTO
TECH

AZ | PROYECTOS
EDUCATIVOS

