



Qué es Python, para que se usa y cómo aprenderlo gratis desde 0

Descripción

Hoy te adentrarás en el universo de Python, un lenguaje de programación que ha conquistado el corazón de empresas y desarrolladores de todo el mundo. Si alguna vez te has preguntado qué es Python, para qué se utiliza y cómo puedes aprenderlo sin gastar un euro, estás en el lugar adecuado.

Además, si estás interesado en aprender Python, te invitamos a explorar nuestro [curso gratis de Python y Django](#), una excelente manera de iniciarte en este lenguaje de programación versátil y poderoso. Además, si estás buscando ampliar tus habilidades en otros lenguajes y tecnologías, también ofrecemos [curso gratis de Desarrollo de Aplicaciones Web con ASP.NET](#), [curso gratis de Programacion.net](#), [curso gratis de Análisis En Código BDD Y TDD](#), [curso gratis de Programación en Visual C++](#), y muchos más. Descubre el apasionante mundo de la programación con nuestros [cursos online gratis de programación](#). ¡Comienza tu viaje hacia el éxito hoy mismo!

Pero, ¿qué lo hace tan especial?

Además de ser gratuito, Python cuenta con una comunidad de desarrolladores tan grande y activa que siempre encontrarás ayuda cuando la necesites. Su versatilidad es asombrosa: desde construir aplicaciones web hasta explorar el vasto mundo de la inteligencia artificial, Python lo hace todo. Y lo mejor de todo, puedes ejecutarlo en cualquier sistema operativo, ya sea Windows, macOS o Linux, sin complicaciones.

En este artículo, desentrañaremos sus misterios, exploraremos sus orígenes, descubriremos por qué es tan popular en el mundo empresarial y, lo más importante, te proporcionaremos recursos gratuitos para que puedas aprenderlo desde cero y convertirte en un maestro de la programación.

Origen de Python

Para comprender verdaderamente la magia detrás de Python, necesitamos hacer un viaje en el tiempo hacia finales de los años 80 y principios de los 90. En ese contexto histórico, nació nuestro querido lenguaje de programación.

Imagina el año 1989, un momento en el que los ordenadores no eran tan poderosos como los de hoy, pero la chispa de la innovación ardía con fuerza. [Guido van Rossum](#), un brillante mente del [Centro de Investigación CWI en los Países Bajos](#), decide iniciar un proyecto que cambiaría el mundo de la programación: Python.

Su germen se encuentra en el proyecto ABC, un lenguaje de programación desarrollado por el equipo de van Rossum en el CWI. Sin embargo, el proyecto ABC tenía limitaciones, y Guido se propuso superarlas. Con una mezcla de inspiración y determinación, comenzó a darle vida a Python, un proyecto que inicialmente fue un pasatiempo, pero que pronto se convertiría en una revolución en el mundo de la programación.

El nombre «Python» no proviene de la serpiente, sino de la afición de Guido van Rossum al grupo de comedia británico [Monty Python](#). La idea era que el lenguaje fuera divertido, fácil de usar y aprendizaje ameno, una filosofía que ha perdurado a lo largo del tiempo.

Aunque en sus primeros días no atrajo la atención que merecía debido a las limitaciones de hardware de la época, el avance tecnológico en las décadas siguientes allanó el camino para que brillara con todo su esplendor.

¿Qué es Python y por qué es interesante?

Python es mucho más que un simple lenguaje de programación; es una herramienta poderosa que ha conquistado el corazón de desarrolladores y empresas de todo el mundo. Veamos qué lo hace tan especial:

DEFINICIÓN

Python es un lenguaje de programación interpretado, lo que significa que no necesita ser compilado antes de ejecutarlo. Esta característica lo hace extremadamente flexible y fácil de aprender.

Al ser interpretado, puedes escribir y probar tu código línea por línea sin tener que preocuparte por procesos complicados de compilación. Esto facilita el proceso de desarrollo y permite una mayor agilidad en la escritura y depuración del código.

Es un lenguaje de programación ampliamente utilizado en una variedad de aplicaciones, incluyendo aplicaciones web, desarrollo de software, ciencia de datos y machine learning (ML).

Los desarrolladores lo eligen debido a su eficiencia y facilidad de aprendizaje. Gracias a su naturaleza interpretada, Python permite a los programadores escribir y probar código de manera rápida y eficaz, sin la necesidad de procesos complicados de compilación. Además, es conocido por su versatilidad, ya que se puede ejecutar en una amplia gama de plataformas, lo que lo hace ideal para proyectos multiplataforma.

TABLA CARACTERÍSTICAS

Característica	Descripción
Legibilidad	Se destaca por su sintaxis clara y legible, lo que facilita la comprensión del código incluso para aquellos que están comenzando en la programación.
Gratuidad y código abierto	Es un lenguaje de código abierto y completamente gratuito, lo que significa que cualquiera puede descargarlo, usarlo y contribuir a su desarrollo sin coste alguno.
Amplia comunidad	Python cuenta con una enorme comunidad de desarrolladores en todo el mundo, lo que garantiza un soporte continuo, una gran cantidad de bibliotecas y recursos educativos.
Flexibilidad	Es un lenguaje multiparadigma que permite a los programadores utilizar diferentes estilos de programación, como la programación orientada a objetos, funcional o procedural, según sea necesario.
Versatilidad	Puede ser utilizado en una amplia gama de aplicaciones, desde el desarrollo de aplicaciones web y móviles hasta la inteligencia artificial, el análisis de datos y la automatización de tareas.
Portabilidad	Es compatible con múltiples plataformas, lo que significa que puede ser ejecutado en diversos sistemas operativos como Windows, macOS y Linux sin necesidad de modificar el código.

Ventajas e inconvenientes de Python

Ventajas frente a otros lenguajes

- **Simplicidad y legibilidad:** Se destaca por su sintaxis clara y legible, lo que facilita la escritura y comprensión del código. Esto hace que sea más fácil para los desarrolladores trabajar en proyectos colaborativos y mantener el código a lo largo del tiempo.
- **Gran cantidad de bibliotecas y frameworks:** Cuenta con una amplia variedad de bibliotecas y frameworks que cubren casi todas las áreas de desarrollo, desde la creación de aplicaciones web hasta el análisis de datos y la inteligencia artificial. Esto permite a los desarrolladores aprovechar el trabajo previo y acelerar el desarrollo de sus proyectos.
- **Comunidad activa y soporte:** Tiene una comunidad de desarrolladores extremadamente activa y colaborativa. Esto significa que siempre hay recursos disponibles, como tutoriales, documentación y foros de discusión, para ayudarte a resolver problemas y mejorar tus

habilidades..

- **Portabilidad:** Python es compatible con múltiples plataformas, lo que significa que puedes escribir código en un sistema operativo y ejecutarlo en otro sin realizar cambios significativos. Esto hace que sea fácil desarrollar y desplegar aplicaciones en diferentes entornos.

Inconvenientes

Aunque Python tiene muchas ventajas, también tiene sus desventajas. Uno de los principales inconvenientes es su velocidad de ejecución en comparación con lenguajes compilados como C++ o Java. Debido a su naturaleza interpretada, Python tiende a ser más lento en la ejecución de ciertas operaciones.

Análisis de cómo los avances tecnológicos mitigan estos inconvenientes

Afortunadamente, los avances tecnológicos han mitigado en gran medida este inconveniente. Por ejemplo, la introducción de implementaciones de Python optimizadas, como PyPy y Cython, ha mejorado significativamente el rendimiento de Python en ciertos escenarios.

Además, el desarrollo de bibliotecas y herramientas de optimización, como NumPy y pandas, ha permitido a los desarrolladores realizar operaciones numéricas y de análisis de datos de manera más eficiente en Python.

Finalmente, el crecimiento de la computación en la nube ha proporcionado acceso a recursos informáticos escalables y de alto rendimiento, lo que permite ejecutar aplicaciones Python de manera eficiente, incluso en entornos con grandes cargas de trabajo.

Tabla resumen de Ventajas e Inconvenientes

Ventajas	Inconvenientes
Legibilidad del código	Velocidad de ejecución inferior a lenguajes compilados como C++
Gratuito y código abierto	Limitaciones en el desarrollo de aplicaciones de tiempo real
Gran comunidad de desarrolladores	Problemas de escalabilidad en aplicaciones de gran envergadura
Flexibilidad multiparadigma	Dependencia de la indentación puede dificultar la lectura para algunos programadores
Amplia gama de aplicaciones	Mayor consumo de recursos en comparación con lenguajes de bajo nivel
Portabilidad entre plataformas	Algunas bibliotecas pueden tener rendimiento subóptimo o falta de soporte

Principios de diseño y el Zen de Python

Para entender verdaderamente la esencia de Python, debemos sumergirnos en su filosofía de diseño, que se encuentra encapsulada en el Zen de Python. Estos principios son como las reglas no escritas

que guían a los desarrolladores en el camino hacia un código limpio, legible y eficiente.

Explicación de los principios de diseño del lenguaje

El Zen de Python es una colección de 19 aforismos que resumen los principios fundamentales de Python. Estos aforismos son como pequeñas joyas de sabiduría que nos recuerdan las mejores prácticas de programación y nos inspiran a escribir código de la manera más elegante y efectiva posible.

Algunos de estos principios incluyen:

- **Explícito es mejor que implícito:** Es mejor ser claro y directo en lugar de dejar que el código sea ambiguo o confuso.
- **Simple es mejor que complejo:** La simplicidad es clave en el diseño de software. Es preferible un diseño simple y fácil de entender que uno complicado y difícil de mantener.
- **Complejo es mejor que complicado:** Aunque a veces es necesario manejar situaciones complejas, es importante evitar la complicación innecesaria. El código debe ser complejo solo cuando sea necesario, pero nunca complicado.
- **Si la implementación es difícil de explicar, es una mala idea:** Un diseño claro y comprensible es esencial para facilitar la colaboración y el mantenimiento del código a lo largo del tiempo.

Relación con las mejores prácticas en desarrollo de software

Los principios del Zen de Python están estrechamente relacionados con las mejores prácticas en desarrollo de software. Al seguir estos principios, los desarrolladores pueden escribir código que sea fácil de entender, mantener y depurar. Esto conduce a un código más robusto, flexible y escalable, que es fundamental para el éxito de cualquier proyecto de software.

Además, al seguir los principios del Zen de Python, los desarrolladores fomentan una cultura de colaboración, transparencia y respeto en sus equipos de trabajo. Esto promueve una mayor eficiencia y productividad en el desarrollo de software, ya que todos están alineados en torno a los mismos valores y objetivos.

Iniciarse en Python

Instalar Python

Para instalar Python, sigue estos pasos:

1. **Windows:** Descarga el instalador desde [este enlace](#) y ejecútalo. Asegúrate de marcar la opción «Incluir Python en Path» durante la instalación para facilitar su uso desde la línea de comandos.
2. **Linux:** En muchas distribuciones, Python ya viene preinstalado. Si no es así, puedes instalarlo desde el gestor de paquetes de tu distribución. Por ejemplo, en Ubuntu puedes usar el comando `sudo apt-get install python3`.
3. **Resto de sistemas operativos:** Consulta la guía de instalación disponible en [este enlace](#) para obtener instrucciones específicas para tu sistema.

Para ejecutar código en Python, tienes dos opciones principales:

1. Lanzar el intérprete de Python desde la línea de comandos, donde puedes escribir y ejecutar código directamente.
2. Crear un archivo de texto con extensión «.py» y escribir el código de tu script en él. Luego, puedes ejecutar el script haciendo doble clic sobre él.

Ejemplos de código

```
# Imprimir un texto en pantalla
print('Hola mundo')

# Recibir un texto del usuario
texto = input('Introduzca un texto: ')

# Recibir un número del usuario
num = int(input('Introduzca un número: '))

# Estructura condicional para determinar si un número es par o impar
if num % 2 == 0:
    print('Par')
else:
    print('Impar')

# Bucle for para imprimir los primeros 20 números
for i in range(20):
    print(i, end=" ") # Imprimir número sin salto de línea
print() # Línea vacía

# Bucle foreach para imprimir elementos de una lista
for i in [1, 5, 7]:
    print(i, end=" ") # Imprimir número sin salto de línea
print() # Línea vacía

# Bucle foreach para imprimir caracteres de un texto
for i in "TEXTO":
    print(i)

# Bucle while para imprimir los primeros 20 números
i = 1
while i <= 20:
    print(i, end=" ")
    i += 1
```

```
# Definir una función para calcular el máximo de dos números
def maximo(n1, n2):
    if n1 < n2:
        return n2
    elif n2 < n1:
        return n1
    else:
        return n1

# Usar la función para calcular el máximo entre 100 y 50
print(maximo(100, 50))
```

¿Qué son las bibliotecas de Python?

Una biblioteca es una colección de códigos utilizados con frecuencia que los desarrolladores pueden incluir en sus programas de Python para evitar tener que escribir el código desde cero. **Python** incluye por defecto la *biblioteca estándar*, que contiene una gran cantidad de funciones reutilizables. Además, existen más de 137,000 bibliotecas de Python disponibles para diversas aplicaciones, como el desarrollo web, la ciencia de datos y el machine learning (ML).

¿Cuáles son las bibliotecas de Python más populares?

Matplotlib

Los desarrolladores utilizan **Matplotlib** para trazar datos en gráficos de alta calidad en dos y tres dimensiones (2D y 3D). Esta biblioteca es muy utilizada en aplicaciones científicas, permitiendo visualizar datos en diferentes tipos de gráficos, como de barras y de líneas.

Pandas

Pandas proporciona estructuras de datos optimizadas y flexibles para manipular datos de serie temporal y estructurados, como tablas y matrices. Es ampliamente utilizada en tareas de ciencia de datos, análisis de datos y ML.

NumPy

NumPy es una conocida biblioteca utilizada para crear y administrar matrices, realizar operaciones de álgebra lineal y manipular formas lógicas con facilidad. Es compatible con la integración en múltiples lenguajes, como C y C++.

Requests

La biblioteca **Requests** proporciona funciones útiles para el desarrollo web, permitiendo enviar solicitudes HTTP, agregar encabezados, parámetros de URL y datos, entre otras tareas.

OpenCV-Python

OpenCV-Python es utilizada para procesar imágenes en aplicaciones de visión artificial, ofreciendo funciones para tareas como lectura y escritura de imágenes, creación de entornos 3D a partir de imágenes 2D, y análisis de imágenes de video.

Keras

Keras es la biblioteca de redes neuronales profundas de Python, con soporte para el procesamiento y visualización de datos, y ofrece una estructura modular que permite la creación de aplicaciones innovadoras con flexibilidad.

¿Qué son los marcos de Python?

Un marco de Python es una colección de paquetes y módulos que los desarrolladores pueden utilizar para crear aplicaciones de Python más rápido. Un módulo es un conjunto de código relacionado, y un paquete es un conjunto de módulos. Los marcos de Python permiten a los desarrolladores centrarse en la lógica de la aplicación sin tener que preocuparse por los detalles de bajo nivel.

Python tiene dos tipos principales de marcos:

- El marco de pila completa, que incluye casi todo lo necesario para crear una aplicación grande.
- El micromarco, que proporciona funcionalidades mínimas para crear aplicaciones simples y también ofrece extensiones para funciones más sofisticadas si es necesario.

¿Cuáles son los marcos de Python más conocidos?

Django

Django es uno de los marcos web de pila completa más utilizados para el desarrollo de aplicaciones web a gran escala. Proporciona numerosas características útiles, como un servidor web integrado, un motor de plantillas y mecanismos de seguridad.

Flask

Flask es un micromarco utilizado para el desarrollo de aplicaciones web pequeñas. Cuenta con un sólido soporte de la comunidad, una documentación bien escrita y características como un motor de plantillas y un servidor web integrado.

TurboGears

TurboGears es un marco diseñado para crear aplicaciones web de manera rápida y sencilla, ofreciendo características como una estructura específica de tabla de base de datos y herramientas para la creación y gestión de proyectos.

Apache MXNet

Apache MXNet es un marco de aprendizaje profundo rápido, flexible y escalable utilizado para crear prototipos de investigación y aplicaciones de aprendizaje profundo. Ofrece un completo conjunto de herramientas y bibliotecas para brindar soporte al desarrollo en múltiples lenguajes de programación.

PyTorch

PyTorch es un marco para el machine learning creado sobre la biblioteca Torch, utilizado en aplicaciones de NLP, robótica y visión artificial para encontrar información significativa en imágenes y videos, y ejecutar aplicaciones en CPU y GPU.

¿Qué son los IDE de Python?

Un IDE (entorno de desarrollo integrado) es un software que proporciona a los desarrolladores todas las herramientas necesarias para escribir, editar, probar y depurar código en un solo lugar, lo que mejora la eficiencia y la productividad del desarrollo de software.

¿Cuáles son los IDE de Python más conocidos?

PyCharm

PyCharm es un IDE desarrollado por JetBrains que ofrece una edición comunitaria gratuita y una edición profesional de pago. Es adecuado para proyectos pequeños y grandes y cuenta con características como completado automático de código, gestión de errores, soporte para frameworks web como Django y Flask, y herramientas científicas como Matplotlib y NumPy.

IDLE

IDLE es el IDE de Python instalado por defecto y desarrollado solo con Python a través de la biblioteca de GUI Tkinter. Ofrece una ventana de shell para ejecutar comandos, un editor de texto con resaltado de sintaxis y autocompletado, y un depurador básico.

Spyder

Spyder es un IDE de código abierto especialmente diseñado para científicos y analistas de datos. Ofrece un editor completo de código, una consola interactiva de IPython, herramientas de depuración y bibliotecas científicas como Matplotlib y NumPy.

Atom

Atom es un editor gratuito desarrollado por GitHub que también puede utilizarse como IDE para Python. Ofrece soporte para varios sistemas operativos, instalación sencilla de nuevos paquetes, autocompletado de código, búsqueda de archivos y proyectos, y personalización de la interfaz.

Aplicaciones empresariales de Python

Python no solo es un lenguaje de programación popular entre los desarrolladores individuales, sino que también es ampliamente utilizado por empresas líderes en una variedad de sectores. Aquí tienes algunos ejemplos de empresas destacadas que utilizan Python en sus operaciones:

Python en Google: Impulso Tecnológico en el Gigante de Internet

En el vasto universo de la tecnología, hay un lenguaje de programación que destaca como un recurso invaluable para gigantes como Google. Nos referimos a Python, un lenguaje versátil y poderoso que ha encontrado un hogar en una variedad de productos y servicios emblemáticos de Google.

Google, una de las empresas más innovadoras del mundo, utiliza Python en una variedad de sus plataformas líderes, como YouTube, Google Search y Google Cloud Platform. Pero, ¿qué hace que Python sea tan atractivo para Google? La respuesta radica en su capacidad para abordar una amplia gama de necesidades, desde el desarrollo de aplicaciones web hasta tareas críticas de backend y análisis de datos.

La flexibilidad de Python lo convierte en el compañero ideal para enfrentar los desafíos tecnológicos que Google encuentra en su día a día. Su sintaxis clara y legible, combinada con una amplia gama de bibliotecas y herramientas, lo hacen ideal para el rápido desarrollo y la escalabilidad que demanda una empresa de la magnitud de Google.

en Netflix: Impulsando la Experiencia del Usuario a Nuevas Alturas

En el mundo del entretenimiento en línea, Netflix se destaca como un pionero en la entrega de contenido personalizado y experiencias de usuario excepcionales. Detrás de su plataforma innovadora se encuentra Python, un lenguaje de programación que desempeña un papel fundamental en la infraestructura tecnológica de Netflix.

Netflix utiliza Python en diversos aspectos de su operación, desde la gestión de contenidos hasta la personalización de recomendaciones. Python es especialmente útil para procesar grandes volúmenes de datos, una tarea crítica en una plataforma que maneja una vasta biblioteca de películas y programas de televisión.

Una de las áreas donde Python brilla en Netflix es en la optimización de los algoritmos de recomendación. Gracias a la flexibilidad y eficiencia de Python, Netflix puede analizar el comportamiento de los usuarios y ofrecer recomendaciones personalizadas que mejoren la experiencia de cada usuario.

en Facebook: Potenciando la Innovación en la Red Social más Grande del Mundo

En el corazón de la red social más grande del mundo, Facebook, se encuentra Python, un lenguaje de programación versátil y poderoso que impulsa la innovación en una variedad de áreas clave.

Facebook utiliza Python en una amplia gama de proyectos y aplicaciones, desde el desarrollo de aplicaciones web hasta la investigación de inteligencia artificial. Uno de los proyectos emblemáticos que utiliza Python en Facebook es la plataforma de publicidad, donde Python ayuda a gestionar y optimizar las campañas de publicidad para millones de anunciantes en todo el mundo.

Además, Python es fundamental en la infraestructura de backend de Facebook, donde se encarga de manejar grandes volúmenes de datos y garantizar la escalabilidad y fiabilidad de la plataforma. También se utiliza en el análisis de datos para la toma de decisiones estratégicas, permitiendo a Facebook comprender mejor el comportamiento de sus usuarios y adaptarse rápidamente a las tendencias del mercado.

Python en Goldman Sachs: Potenciando la Eficiencia y Precisión en el Sector Financiero

En el competitivo mundo del sector financiero, empresas como Goldman Sachs se destacan por su enfoque en la innovación y la eficiencia. En este contexto, Python se ha convertido en una herramienta esencial para abordar una variedad de desafíos y aprovechar oportunidades en el mercado.

Goldman Sachs utiliza Python en una amplia gama de propósitos, desde el análisis de datos hasta el modelado financiero y el desarrollo de aplicaciones internas. Python ofrece una combinación única de flexibilidad y potencia que lo hace ideal para abordar las complejidades del mundo financiero.

Una de las áreas donde Python brilla en Goldman Sachs es en el análisis de riesgos. Gracias a su capacidad para manejar grandes volúmenes de datos y ejecutar algoritmos sofisticados, Python ayuda a Goldman Sachs a evaluar y mitigar los riesgos financieros de manera eficiente y precisa.

Además, Python es fundamental en la automatización de tareas repetitivas, lo que permite a Goldman Sachs optimizar sus operaciones y reducir el tiempo y los recursos necesarios para realizar tareas manuales.

Python en J.P. Morgan: Potenciando la Excelencia en el Sector Financiero

En el dinámico mundo de las finanzas, J.P. Morgan se destaca como una institución líder que busca constantemente innovar y mejorar sus operaciones. Python ha surgido como una herramienta fundamental en este proceso de transformación, permitiendo a J.P. Morgan abordar una variedad de desafíos y aprovechar oportunidades en el mercado.

J.P. Morgan utiliza Python en áreas críticas como el trading algorítmico, la gestión de riesgos y el análisis de datos financieros. La flexibilidad y potencia de Python lo hacen ideal para desarrollar modelos predictivos que ayudan a J.P. Morgan a anticipar tendencias del mercado y tomar decisiones

informadas.

Además, Python se utiliza para optimizar estrategias de inversión, permitiendo a J.P. Morgan identificar oportunidades de inversión lucrativas y maximizar los rendimientos para sus clientes.

Otro uso importante de Python en J.P. Morgan es mejorar la eficiencia operativa. Python permite automatizar tareas repetitivas y procesos manuales, lo que ayuda a J.P. Morgan a reducir costos y tiempos de ejecución.

en PayPal: Impulsando la Seguridad y la Escalabilidad en Pagos en Línea

PayPal se destaca como una plataforma líder que se esfuerza por ofrecer seguridad, confiabilidad y una experiencia fluida para sus usuarios. En este escenario, Python desempeña un papel fundamental al impulsar la infraestructura tecnológica detrás de la plataforma de PayPal.

PayPal utiliza Python en una variedad de áreas clave, desde la gestión de transacciones hasta la detección de fraudes y la optimización de la experiencia del usuario. La versatilidad y potencia de Python lo hacen ideal para abordar los desafíos únicos asociados con el procesamiento de millones de transacciones diarias en un entorno en línea.

Una de las áreas donde Python brilla en PayPal es en el desarrollo de aplicaciones web seguras y escalables. Python proporciona las herramientas necesarias para crear aplicaciones robustas que protegen la información confidencial de los usuarios y garantizan una experiencia de pago sin problemas.

Además, Python es utilizado para la detección de fraudes, ayudando a PayPal a identificar y prevenir actividades fraudulentas en tiempo real. Python permite analizar grandes volúmenes de datos y aplicar algoritmos avanzados para identificar patrones sospechosos y tomar medidas rápidas para proteger a los usuarios.

Cómo aprender Python gratis

¿Te entusiasma aprender Python y Django de manera gratuita? ¡Estás de suerte! Existen recursos educativos de calidad que te permitirán sumergirte en el fascinante mundo de la programación. Uno de estos recursos es el [curso gratuito de Python y Django](#) ofrecido por Impulso06, con una duración de 40 horas. Aquí tienes un vistazo al temario del curso:

1. INTRODUCCIÓN

- 1.1. Introducción a Python y Django
- 1.2. Comenzando con Python
- 1.3. Comenzando con Django

2. DJANGO Y FORMULARIOS

3. DJANGO. PLANTILLAS AVANZADAS Y JQUERY

4. ACCESO A DATOS Y ACTIVE RECORD

5. JAVASCRIPT Y DOM

6. FORMULARIOS Y VALIDACIONES

7. MODELOS Y RELACIONES

8. ADMINISTRACIÓN Y SEGURIDAD
9. SESIONES
10. DESPLIEGUES

Este curso te guiará desde los fundamentos hasta temas avanzados, brindándote una comprensión completa de Python y Django.

Recursos disponibles para aprenderlo de forma gratuita

Además de cursos específicos, existen varias plataformas educativas, tutoriales en línea y comunidades de aprendizaje que te ofrecen la oportunidad de aprender Python sin costo alguno. Algunas opciones incluyen:

- [Codecademy](#): Un enfoque interactivo para aprender Python.
- [Documentación oficial de Python](#): Una excelente fuente para comprender la sintaxis y funcionalidades del lenguaje.
- [W3Schools Python Tutorial](#): Tutoriales prácticos para aprender Python paso a paso.
- [Real Python](#): Artículos, tutoriales y videos para desarrolladores de Python de todos los niveles.
- [Stack Overflow](#): Una comunidad donde puedes hacer preguntas y aprender de otros desarrolladores.

Con estos recursos, tendrás todo lo que necesitas para comenzar tu viaje en el mundo de Python de manera gratuita y efectiva. ¡A disfrutar del aprendizaje!

Conclusiones Qué es Python, para que se usa y cómo aprenderlo gratis desde 0

Python se ha convertido en mucho más que un simple lenguaje de programación; es una herramienta poderosa que impulsa la innovación y el progreso en empresas de todo el mundo. A lo largo de este artículo, hemos explorado los fundamentos de Python, desde sus humildes orígenes hasta su uso generalizado en una variedad de campos empresariales.

Desde su nacimiento en los años 80 y 90, Python ha evolucionado para convertirse en un pilar fundamental en el mundo de la tecnología. Su sintaxis clara y legible, su versatilidad como lenguaje multiparadigma y su gran comunidad de desarrolladores son solo algunas de las razones por las que tantas empresas confían en Python para impulsar sus operaciones.

A pesar de sus ventajas, Python no está exento de desafíos. La velocidad de ejecución en comparación con lenguajes compilados es uno de los principales obstáculos, pero los avances tecnológicos están ayudando a mitigar este inconveniente y a hacer que Python sea una opción viable incluso en entornos con grandes cargas de trabajo.

En última instancia, el carácter gratuito, flexible y seguro de Python, respaldado por una comunidad activa y colaborativa, lo convierte en una apuesta segura en el proceso de digitalización de las empresas. A medida que el mundo empresarial avanza hacia un futuro cada vez más digital, tener conocimientos de Python se vuelve no solo una ventaja, sino una necesidad para aquellos que desean mantenerse relevantes y competitivos en la carrera tecnológica sin precedentes que estamos

viviendo.

Impulso06