**Python là gì?**

Python là một ngôn ngữ lập trình được sử dụng rộng rãi trong các ứng dụng web, phát triển phần mềm, khoa học dữ liệu và máy học (ML). Các nhà phát triển sử dụng Python vì nó hiệu quả, dễ học và có thể chạy trên nhiều nền tảng khác nhau. Phần mềm Python được tải xuống miễn phí, tích hợp tốt với tất cả các loại hệ thống và tăng tốc độ phát triển.

**Python mang lại những lợi ích gì?**

Những lợi ích của Python bao gồm:

• Các nhà phát triển có thể dễ dàng đọc và hiểu một chương trình Python vì ngôn ngữ này có cú pháp cơ bản giống tiếng Anh.

• Python giúp cải thiện năng suất làm việc của các nhà phát triển vì so với những ngôn ngữ khác, họ có thể sử dụng ít dòng mã hơn để viết một chương trình Python.

• Python có một thư viện tiêu chuẩn lớn, chứa nhiều dòng mã có thể tái sử dụng cho hầu hết mọi tác vụ. Nhờ đó, các nhà phát triển sẽ không cần phải viết mã từ đầu.

• Các nhà phát triển có thể dễ dàng sử dụng Python với các ngôn ngữ lập trình phổ biến khác như Java, C và C++.

• Cộng đồng Python tích cực hoạt động bao gồm hàng triệu nhà phát triển nhiệt tình hỗ trợ trên toàn thế giới. Nếu gặp phải vấn đề, bạn sẽ có thể nhận được sự hỗ trợ nhanh chóng từ cộng đồng.

• Trên Internet có rất nhiều tài nguyên hữu ích nếu bạn muốn học Python. Ví dụ: bạn có thể dễ dàng tìm thấy video, chỉ dẫn, tài liệu và hướng dẫn dành cho nhà phát triển.

• Python có thể được sử dụng trên nhiều hệ điều hành máy tính khác nhau, chẳng hạn như Windows, macOS, Linux và Unix.

Python được sử dụng như thế nào?

Ngôn ngữ Python được sử dụng nhiều trong lĩnh vực phát triển ứng dụng, bao gồm những ví dụ sau:

Phát triển web phía máy chủ

Phát triển web phía máy chủ bao gồm những hàm backend phức tạp mà các trang web thực hiện để hiển thị thông tin cho người dùng. Ví dụ: các trang web phải tương tác với cơ sở dữ liệu, giao tiếp với các trang web khác và bảo vệ dữ liệu khi truyền qua mạng.

Python hữu ích trong việc lập trình mã phía máy chủ bởi vì ngôn ngữ này cung cấp nhiều thư viện bao gồm mã viết sẵn cho các hàm backend phức tạp. Các nhà phát triển cũng sử dụng một loạt các khung Python cung cấp tất cả những công cụ cần thiết để xây dựng ứng dụng web một cách nhanh chóng và dễ dàng hơn. Ví dụ: các nhà phát triển có thể tạo ứng dụng web khung trong nháy mắt bởi vì họ không cần phải lập trình nó từ đầu. Sau đó, họ có thể kiểm tra ứng dụng web này bằng cách sử dụng các công cụ kiểm thử của khung, mà không cần phụ thuộc vào những công cụ kiểm thử bên ngoài.

**Tự động hóa bằng các tập lệnh Python**

Ngôn ngữ tập lệnh là một ngôn ngữ lập trình tự động hóa các tác vụ mà thường được con người thực hiện. Các lập trình viên thường xuyên sử dụng các tập lệnh Python để tự động hóa nhiều tác vụ hàng ngày như:

• Đổi tên một số lượng lớn tệp cùng lúc

• Chuyển đổi một tệp sang một loại tệp khác

• Loại bỏ các từ trùng lặp trong tệp văn bản

• Thực hiện các phép tính toán cơ bản

• Gửi email

• Tải xuống nội dung

• Thực hiện phân tích nhật ký cơ bản

• Tìm kiếm lỗi trong nhiều tệp

Khoa học dữ liệu và máy học

Khoa học dữ liệu trích xuất thông tin quý giá từ dữ liệu và máy học (ML) dạy máy tính tự động học hỏi từ dữ liệu và đưa ra các dự đoán chính xác. Các nhà khoa học dữ liệu sử dụng Python cho các tác vụ khoa học dữ liệu sau:

• Sửa và loại bỏ dữ liệu không chính xác, hay còn được gọi là làm sạch dữ liệu

• Trích xuất và chọn lọc các đặc điểm đa dạng của dữ liệu

• Ghi nhãn dữ liệu gán tên có ý nghĩa cho dữ liệu

• Tìm các số liệu thống kê khác nhau từ dữ liệu

• Trực quan hóa dữ liệu bằng cách sử dụng các biểu đồ và đồ thị, chẳng hạn như biểu đồ đường, biểu đồ cột, biểu đồ tần suất và biểu đồ tròn

Các nhà khoa học dữ liệu sử dụng những thư viện ML của Python để đào tạo các mô hình ML và xây dựng các công cụ phân loại giúp phân loại dữ liệu một cách chính xác. Các chuyên gia từ nhiều lĩnh vực sử dụng những công cụ phân loại dựa trên Python để thực hiện các tác vụ phân loại, chẳng hạn như phân loại hình ảnh, văn bản cũng như lưu lượng truy cập mạng, nhận dạng giọng nói và nhận diện khuôn mặt. Các nhà khoa học dữ liệu cũng sử dụng Python cho deep learning, một kỹ thuật ML nâng cao.

Phát triển phần mềm

Các nhà phát triển phần mềm thường sử dụng Python cho những tác vụ phát triển và ứng dụng phần mềm khác nhau, chẳng hạn như:

• Theo dõi lỗi trong mã của phần mềm

• Tự động xây dựng phần mềm

• Đảm nhận quản lý dự án phần mềm

• Phát triển nguyên mẫu phần mềm

• Phát triển các ứng dụng máy tính bằng cách sử dụng những thư viện Giao diện đồ họa người dùng (GUI)

• Phát triển từ các trò chơi văn bản đơn giản cho đến những trò chơi điện tử phức tạp

Tự động hóa kiểm thử phần mềm

Kiểm thử phần mềm là quy trình kiểm tra xem kết quả thực tế từ phần mềm có khớp với kết quả mong đợi không để đảm bảo rằng phần mềm không có lỗi.

• Các nhà phát triển sử dụng khung kiểm thử đơn vị Python, chẳng hạn như Unittest, Robot và PyUnit, để kiểm thử các hàm do họ viết.

• Các kỹ sư kiểm thử phần mềm sử dụng Python để viết các trường hợp kiểm thử cho nhiều tình huống khác nhau. Ví dụ: họ sử dụng ngôn ngữ này để kiểm thử giao diện người dùng của một ứng dụng web, nhiều thành phần của phần mềm và những tính năng mới.

Các nhà phát triển có thể sử dụng một số công cụ để tự động chạy tập lệnh kiểm thử. Những công cụ này có tên gọi là công cụ Tích hợp liên tục/Triển khai liên tục (CI/CD). Các kỹ sư kiểm thử phần mềm cũng như những nhà phát triển sử dụng các công cụ CI/CD như Travis CI và Jenkins để tự động hóa quy trình kiểm thử. Công cụ CI/CD tự động chạy các tập lệnh kiểm thử Python và báo cáo kết quả kiểm thử bất kỳ khi nào nhà phát triển thêm vào những dòng mã mới.

**Python có lịch sử như thế nào?**

Guido Van Rossum, một lập trình viên máy tính ở Hà Lan, đã tạo ra Python. Ông bắt đầu phát triển ngôn ngữ này vào năm 1989 tại Centrum Wiskunde & Informatica (CWI), ban đầu chỉ là một dự án tiêu khiển để giúp ông giết thời gian trong dịp Giáng sinh. Tên gọi của ngôn ngữ này được lấy cảm hứng từ chương trình truyền hình Monty Python's Flying Circus của đài BBC vì Guido Van Rossum là một “fan cứng” của chương trình này.

**Lịch sử các phiên bản Python**

• Guido Van Rossum cho ra mắt phiên bản đầu tiên của ngôn ngữ Python (phiên bản 0.9.0) vào năm 1991. Ngôn ngữ này đã bao gồm các tính năng hữu ích như một số kiểu dữ liệu và hàm để xử lý lỗi.

• Python 1.0 đã được ra mắt vào năm 1994 với các hàm mới để dễ dàng xử lý danh sách dữ liệu, chẳng hạn như ánh xạ, lọc và lược bỏ.

• Python 2.0 đã được ra mắt vào ngày 16 tháng 10 năm 2000, với các tính năng hữu ích mới cho lập trình viên, chẳng hạn như hỗ trợ ký tự Unicode và cách xử lý chi tiết một danh sách nhanh chóng hơn.

• Python 3.0 đã được ra mắt vào ngày 3 tháng 12 năm 2008. Phiên bản này bao gồm các tính năng như hàm in và hỗ trợ nhiều hơn cho việc phân chia số và xử lý lỗi.

**Python có những đặc điểm gì?**

Các đặc điểm sau tạo nên sự độc đáo của ngôn ngữ lập trình Python:

Python là một ngôn ngữ thông dịch

Python là một ngôn ngữ thông dịch, điều này nghĩa là ngôn ngữ này trực tiếp chạy từng dòng mã. Nếu có lỗi trong mã chương trình, nó sẽ ngừng chạy. Do đó, lập trình viên có thể nhanh chóng tìm ra lỗi trong đoạn mã.

Python là một ngôn ngữ dễ sử dụng

Python sử dụng từ ngữ giống trong tiếng Anh. Không giống như các ngôn ngữ lập trình khác, Python không sử dụng dấu ngoặc ôm. Thay vào đó, ngôn ngữ này sử dụng thụt đầu dòng.

Python là một ngôn ngữ linh hoạt

Các lập trình viên không cần phải khai báo loại biến khi viết mã bởi vì Python sẽ xác định chúng vào thời điểm chạy. Vì vậy, bạn có thể viết các chương trình Python một cách nhanh chóng hơn.

Python là một ngôn ngữ cấp cao

Python gần gũi với ngôn ngữ con người hơn các ngôn ngữ lập trình khác. Do đó, các lập trình viên không cần phải lo lắng về những chức năng cơ bản của nó như kiến trúc và quản lý bộ nhớ.

Python là một ngôn ngữ lập trình hướng đối tượng

Python coi mọi thứ đều là đối tượng, nhưng ngôn ngữ này cũng hỗ trợ các phương thức lập trình khác như lập trình hàm và lập trình cấu trúc.

**Thư viện Python là gì?**

Thư viện là một tập hợp các mã thường xuyên được sử dụng mà các nhà phát triển có thể bao gồm trong những chương trình Python của họ để không phải lập trình từ đầu. Theo mặc định, Python đi kèm với Thư viện chuẩn, chứa rất nhiều các hàm có thể tái sử dụng. Ngoài ra, hơn 137.000 thư viện Python có sẵn cho các ứng dụng khác nhau, bao gồm phát triển web, khoa học dữ liệu và máy học (ML).

Thư viện Python nào phổ biến nhất?

Matplotlib

Các nhà phát triển sử dụng Matplotlib để hiển thị dữ liệu dưới dạng đồ họa hai và ba chiều (2D và 3D) chất lượng cao. Thư viện này thường được sử dụng trong các ứng dụng khoa học. Với Matplotlib, bạn có thể trực quan hóa dữ liệu bằng cách hiển thị dữ liệu dưới dạng các biểu đồ khác nhau, chẳng hạn như biểu đồ cột và biểu đồ đường. Bạn cũng có thể hiển thị nhiều biểu đồ cùng lúc và các chi tiết đồ họa có thể được di chuyển qua mọi nền tảng.

Pandas

Pandas cung cấp cấu trúc dữ liệu được tối ưu hóa và linh hoạt mà bạn có thể sử dụng để thao tác với dữ liệu chuỗi thời gian và dữ liệu có cấu trúc, chẳng hạn như bảng và nhóm. Ví dụ, bạn có thể sử dụng Pandas để đọc, ghi, hợp nhất, lọc và nhóm dữ liệu. Thư viện này được nhiều người sử dụng cho các tác vụ khoa học dữ liệu, phân tích dữ liệu và ML.

NumPy

NumPy là một thư viện phổ biến mà các nhà phát triển sử dụng để dễ dàng tạo và quản lý nhóm, thao tác với các hình dạng logic và thực hiện các phép toán đại số tuyến tính. NumPy hỗ trợ tích hợp với nhiều ngôn ngữ như C và C ++.

Requests

Thư viện Requests cung cấp nhiều hàm hữu ích cần thiết để phát triển web. Bạn có thể sử dụng thư viện này để gửi các yêu cầu HTTP, bổ sung tiêu đề, thêm tham số URL, bổ sung dữ liệu và thực hiện nhiều tác vụ khác khi giao tiếp với các ứng dụng web.

OpenCV-Python

OpenCV-Python là một thư viện mà các nhà phát triển sử dụng để xử lý hình ảnh cho các ứng dụng thị giác máy tính. Thư viện này cung cấp nhiều hàm cho các tác vụ xử lý hình ảnh như đọc và ghi hình ảnh cùng lúc, xây dựng môi trường 3D từ môi trường 2D cũng như chụp và phân tích hình ảnh từ video.

Keras

Keras là thư viện mạng nơ-ron chuyên sâu của Python với khả năng hỗ trợ tuyệt vời cho việc xử lý dữ liệu, trực quan hóa và hơn thế nữa. Keras hỗ trợ nhiều mạng nơ-ron. Thư viện này có cấu trúc mô-đun mang lại sự linh hoạt cho việc lập trình các ứng dụng sáng tạo.

**Khung Python là gì?**

Khung Python là tập hợp của các gói và mô-đun. Mô-đun là tập hợp của các mã liên quan và gói là tập hợp của các mô-đun. Các nhà phát triển có thể sử dụng khung Python để xây dựng các ứng dụng Python nhanh hơn vì họ không phải lo lắng về các chi tiết cấp thấp như cách giao tiếp trong ứng dụng web hoặc cách Python sẽ làm cho chương trình chạy nhanh hơn. Python có hai loại khung:

• Khung full-stack bao gồm hầu hết mọi thứ cần thiết để xây dựng một ứng dụng lớn.

• Khung micro là một khung cơ bản cung cấp các chức năng tối thiểu để xây dựng những ứng dụng Python đơn giản. Nó cũng cung cấp các tiện ích mở rộng nếu ứng dụng cần các hàm phức tạp hơn.

Khung Python nào phổ biến nhất?

Các nhà phát triển có thể sử dụng nhiều khung Python để giúp hoạt động phát triển của họ hiệu quả hơn, bao gồm các khung sau:

Django

Django là một trong những khung web Python full-stack được sử dụng rộng rãi nhất để phát triển các ứng dụng web trên quy mô lớn. Khung này cung cấp một số tính năng hữu ích, bao gồm một máy chủ web để phát triển và kiểm thử, một công cụ mẫu để xây dựng giao diện người dùng của trang web và nhiều cơ chế bảo mật khác.

Flask

Flask là một khung micro được sử dụng để phát triển các ứng dụng web nhỏ. Các đặc điểm của khung này bao gồm sự hỗ trợ nhiệt tình từ cộng đồng, tài liệu hữu ích, công cụ mẫu, tính năng kiểm thử đơn vị và máy chủ web được tích hợp sẵn. Nó cũng cung cấp các tiện ích mở rộng để hỗ trợ xác thực, các lớp ánh xạ cơ sở dữ liệu và bảo mật web.

TurboGears

TurboGears là một khung được thiết kế để xây dựng các ứng dụng web một cách nhanh chóng và dễ dàng hơn. Sau đây là một số đặc điểm nổi bật của khung này:

• Cấu trúc bảng cơ sở dữ liệu cụ thể

• Các công cụ tạo và quản lý dự án

• Một công cụ mẫu để xây dựng cơ sở dữ liệu

• Một công cụ mẫu để tạo giao diện người dùng

• Các cơ chế bảo mật web

Apache MXNet

Apache MXNet là một khung học sâu nhanh, linh hoạt và có thể điều chỉnh quy mô được các nhà phát triển sử dụng để xây dựng những nguyên mẫu nghiên cứu và ứng dụng học sâu. Khung này hỗ trợ nhiều ngôn ngữ lập trình, bao gồm Java, C++, R và Perl. Nó cung cấp một bộ công cụ cũng như thư viện phong phú để hỗ trợ phát triển. Ví dụ: bạn có thể tìm thấy một cuốn sách tương tác về máy học (ML), bộ công cụ thị giác máy tính và các mô hình học sâu dành cho Xử lý ngôn ngữ tự nhiên (NLP) để xử lý ngôn ngữ tự nhiên, chẳng hạn như văn bản và lời nói.

PyTorch

PyTorch là một khung cho ML được xây dựng dựa trên thư viện Torch, một thư viện ML nguồn mở khác. Các nhà phát triển sử dụng khung này cho những ứng dụng như NLP, robot và thị giác máy tính, tìm kiếm thông tin có ý nghĩa từ hình ảnh và video. Họ cũng sử dụng PyTorch để chạy những ứng dụng đó bằng CPU và GPU.

**Python IDE là gì?**

Môi trường phát triển tích hợp (IDE) là phần mềm cung cấp cho các nhà phát triển công cụ duy nhất họ cần để viết, chỉnh sửa, kiểm tra và gỡ lỗi mã.

**Python IDE nào phổ biến nhất?**

PyCharm

JetBrains, một công ty phát triển công cụ phần mềm tại Séc, đã tạo ra PyCharm. Nó có một phiên bản cộng đồng miễn phí phù hợp với các ứng dụng Python nhỏ và một phiên bản chuyên nghiệp trả phí dành cho việc xây dựng các ứng dụng Python quy mô lớn, với đầy đủ những tính năng sau:

• Tự động hoàn thành và kiểm tra mã

• Xử lý lỗi và sửa lỗi nhanh chóng

• Làm sạch mã mà không thay đổi chức năng

• Hỗ trợ các khung ứng dụng web như Django và Flask

• Hỗ trợ các ngôn ngữ lập trình khác, chẳng hạn như JavaScript, CoffeeScript, TypeScript, AngularJS và Node

• Các công cụ và thư viện khoa học như Matplotlib và NumPy

• Khả năng chạy, gỡ lỗi, kiểm thử và triển khai các ứng dụng trong máy ảo từ xa

• Trình gỡ lỗi để tìm lỗi trong mã, trình kiểm tra để xác định các vấn đề về hiệu suất trong mã và trình chạy thử nghiệm để chạy các bài kiểm thử đơn vị

• Hỗ trợ cơ sở dữ liệu

IDLE

Môi trường phát triển và học hỏi tích hợp (IDLE) là Python IDE được cài đặt theo mặc định. Nó chỉ được phát triển với Python bằng bộ công cụ Tkinter GUI và cung cấp các tính năng sau:

• Hoạt động trên nhiều hệ điều hành như Windows, Unix và macOS

• Cung cấp một cửa sổ shell để chạy các lệnh và hiển thị kết quả

• Cung cấp trình soạn thảo văn bản trên nhiều cửa sổ với khả năng đánh dấu cú pháp mã và hoàn thành mã tự động

• Có trình gỡ lỗi riêng

Spyder

Spyder là một IDE nguồn mở được nhiều nhà khoa học và nhà phân tích dữ liệu sử dụng. Nó cung cấp trải nghiệm phát triển toàn diện với các tính năng phân tích dữ liệu nâng cao, trực quan hóa dữ liệu và gỡ lỗi. Spyder cũng bao gồm các tính năng sau:

• Trình chỉnh sửa mã hỗ trợ nhiều ngôn ngữ

• Bảng điều khiển IPython tương tác

• Trình gỡ lỗi cơ bản

• Thư viện khoa học như Matplotlib, SciPy và NumPy

• Khả năng khám phá các biến trong mã

• Khả năng xem tài liệu trong thời gian thực

Atom

Atom là một trình soạn thảo miễn phí do GitHub phát triển, hỗ trợ viết mã bằng nhiều ngôn ngữ lập trình, bao gồm cả Python. Sử dụng Atom, các nhà phát triển có thể trực tiếp làm việc với GitHub, trang web nơi bạn có thể lưu mã của mình một cách tập trung. Atom cung cấp các tính năng sau:

• Khả năng hoạt động trên nhiều hệ điều hành

• Dễ dàng cài đặt hoặc tạo các gói mới

• Hoàn thành mã tự động nhanh chóng hơn

• Khả năng tìm kiếm tệp và dự án

• Tùy chỉnh giao diện dễ dàng

**SDK Python là gì?**

Bộ công cụ phát triển phần mềm (SDK) là một tập hợp các công cụ phần mềm mà các nhà phát triển có thể sử dụng để tạo ra những ứng dụng phần mềm bằng một ngôn ngữ cụ thể. Hầu hết các SDK đều dành riêng cho những nền tảng phần cứng và hệ điều hành khác nhau. Các SDK Python bao gồm nhiều công cụ như thư viện, mã mẫu và hướng dẫn dành cho nhà phát triển, hữu ích khi lập trình ứng dụng.

**AWS PyCharm là gì?**

Bộ công cụ AWS dành cho PyCharm là tiện ích bổ trợ cho PyCharm IDE, giúp tạo, gỡ lỗi và triển khai các ứng dụng Python trên AWS một cách dễ dàng hơn. Sử dụng Bộ công cụ AWS cho PyCharm, các nhà phát triển có thể dễ dàng bắt đầu phát triển bằng Python. Nó cung cấp một số tính năng hữu ích cho các nhà phát triển, bao gồm hướng dẫn bắt đầu, gỡ lỗi từng bước và triển khai IDE.

**Boto3 trong Python là gì?**

Boto3 là SDK của AWS dành cho Python. Bạn có thể dùng nó để tạo, cấu hình và quản lý các dịch vụ AWS như Đám mây điện toán linh hoạt (EC2) của Amazon, Dịch vụ lưu trữ đơn giản (S3) của Amazon và Amazon DynamoDB. Boto3 cũng cung cấp hai loại API: API cấp thấp và API tài nguyên cho các nhà phát triển.