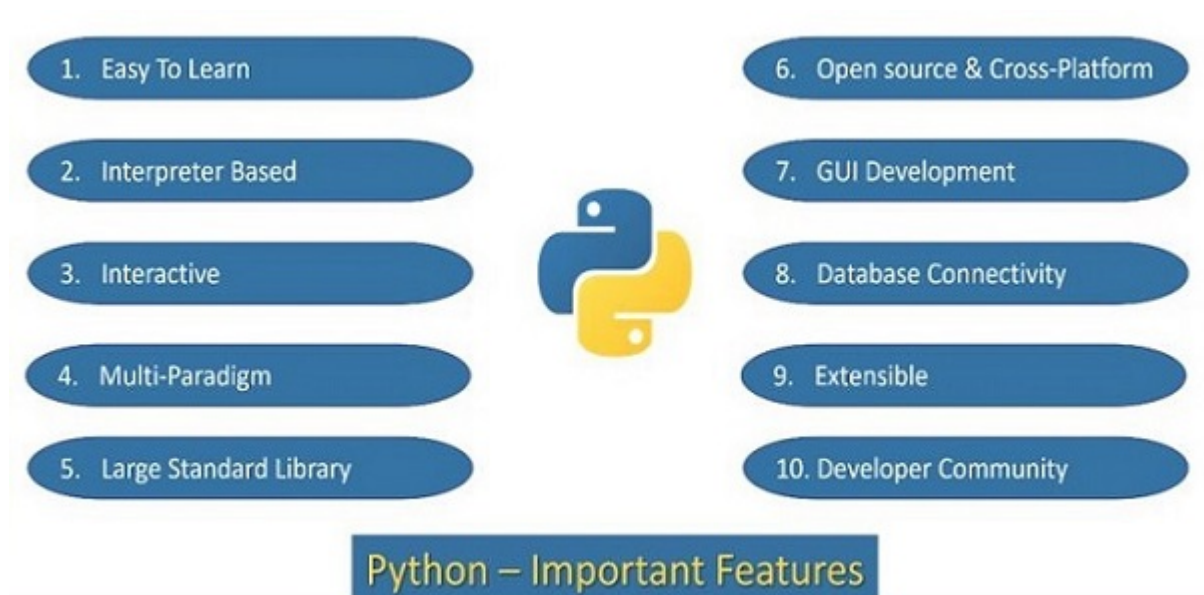


Python - Recursos

Python é uma linguagem de script de alto nível, interpretada, interativa e orientada a objetos, rica em recursos. Este tutorial listará alguns dos recursos importantes do Python:

Recursos do Python

Vamos destacar alguns dos recursos importantes do Python que o tornam amplamente popular. Além desses 10 recursos, há vários outros recursos interessantes que tornam o Python a primeira escolha do desenvolvedor.



A seguir estão os recursos da linguagem de programação Python –

1. Fácil de aprender

Esta é uma das razões mais importantes para a popularidade do Python. Python possui um conjunto limitado de palavras-chave. Seus recursos como **sintaxe** simples, uso de indentação para evitar confusão de colchetes e digitação dinâmica que não requer declaração prévia de variável ajudam um iniciante a aprender Python de forma rápida e fácil.

2. Baseado em Intérprete

As instruções em qualquer linguagem de programação devem ser traduzidas em código de máquina para que o processador as execute. As linguagens de programação são baseadas em compilador ou em interpretador.

No caso de um compilador, é gerada uma versão **em linguagem de máquina** de todo o programa fonte. A conversão falha mesmo se houver uma única instrução errada. Conseqüentemente, o processo de desenvolvimento é tedioso para os iniciantes. As linguagens da família C (incluindo **C** , **C++** , **Java** , **C#** etc) são baseadas em compiladores.

Python é uma linguagem baseada em interpretador. O interpretador pega uma instrução do código-fonte por vez, traduz-a em código de máquina e a executa. As instruções antes da primeira ocorrência de erro são executadas. Com esse recurso, é mais fácil depurar o programa e, portanto, é útil para o programador iniciante ganhar confiança gradativamente. Python, portanto, é uma linguagem amigável para iniciantes.

3. Interativo

A distribuição padrão do Python vem com um shell interativo que funciona segundo o princípio REPL (Leitura – Avaliação – Impressão – Loop). O shell apresenta um prompt Python `>>>`. Você pode digitar qualquer expressão Python válida e pressionar Enter. O interpretador Python retorna imediatamente a resposta e o prompt volta para ler a próxima expressão.

```
>>> 2*3+1
7
>>> print ("Hello World")
Hello World
```

O modo interativo é especialmente útil para se familiarizar com uma biblioteca e testar sua funcionalidade. Você pode experimentar pequenos trechos de código no modo interativo antes de escrever um programa.

4. MultiParadigma

Python é uma linguagem totalmente **orientada a objetos** . Tudo em um programa Python é um **objeto** . No entanto, Python encapsula convenientemente sua orientação a objetos para ser usada como uma linguagem imperativa ou processual – como C. Python também fornece certas funcionalidades que se assemelham à programação funcional. Além disso, certas ferramentas de terceiros foram desenvolvidas para apoiar outros paradigmas de programação, como programação orientada a aspectos e lógica.

5. Biblioteca Padrão

Embora tenha poucas palavras-chave (apenas trinta e cinco), o software Python é distribuído com uma biblioteca padrão composta por um grande número de módulos e pacotes. Assim, o Python tem suporte imediato para necessidades de programação, como serialização, compactação de dados, manipulação de dados da Internet e muito mais. Python é conhecido por sua abordagem com baterias incluídas.

6. Código aberto e plataforma cruzada

A distribuição padrão do Python pode ser baixada em <https://www.python.org/downloads/> sem quaisquer restrições. Você pode baixar binários pré-compilados para várias plataformas de sistema operacional. Além disso, o código-fonte também está disponível gratuitamente, por isso está na categoria de código aberto.

O software Python (junto com a documentação) é distribuído sob a licença Python Software Foundation. É uma licença de software permissiva no estilo BSD e compatível com GNU GPL (General Public License).

Python é uma linguagem multiplataforma. Binários pré-compilados estão disponíveis para uso em diversas plataformas de sistemas operacionais como [Windows](#) , [Linux](#) , [Mac OS](#) , [Android OS](#) . A implementação de referência do Python é chamada CPython e é escrita em C. Você pode baixar o código-fonte e compilá-lo para sua plataforma de sistema operacional.

Um programa Python é primeiro compilado em um código de bytes independente de plataforma intermediária. A máquina virtual dentro do interpretador executa o código de bytes. Esse comportamento torna o Python uma linguagem multiplataforma e, portanto, um programa Python pode ser facilmente portado de uma plataforma de sistema operacional para outra.

7. Aplicativos GUI

A distribuição padrão do Python possui uma excelente biblioteca gráfica chamada TKinter. É uma porta Python para o kit de ferramentas GUI muito popular chamado TCL/Tk. Você pode construir aplicativos GUI atraentes e fáceis de usar em Python. Os kits de ferramentas GUI geralmente são escritos em C/C++. Muitos deles foram portados para Python. Exemplos são [PyQt](#) , [WxWidgets](#) , [PySimpleGUI](#) etc.

8. Conectividade de banco de dados

Quase qualquer tipo de banco de dados pode ser usado como backend com o aplicativo Python. DB-API é um conjunto de especificações para software de driver de banco de dados para permitir que o Python se comunique com um banco de

dados relacional. Com muitas bibliotecas de terceiros, o Python também pode funcionar com bancos de dados NoSQL, como o **MongoDB** .

9. Extensível

O termo extensibilidade implica a capacidade de adicionar novos recursos ou modificar recursos existentes. Conforme afirmado anteriormente, CPython (que é a implementação de referência do Python) é escrito em C. Portanto, é possível escrever facilmente módulos/bibliotecas em C e incorporá-los na biblioteca padrão. Existem outras implementações de Python, como Jython (escrito em Java) e **IPython** (escrito em C#). Portanto, é possível escrever e mesclar novas funcionalidades nessas implementações com Java e C# respectivamente.

10. Comunidade ativa de desenvolvedores

Como resultado da popularidade do Python e da natureza de código aberto, um grande número de desenvolvedores Python frequentemente interage com fóruns e conferências online. A Python Software Foundation também possui uma base significativa de membros, envolvidos na missão da organização de " **Promover, Proteger e Avançar a Linguagem de Programação Python** " .

Python também desfruta de um apoio institucional significativo. As principais empresas de TI Google, Microsoft e Meta contribuem imensamente preparando documentação e outros recursos.

Mais recursos do Python

Além dos recursos mencionados acima, Python tem outra grande lista de bons recursos, poucos estão listados abaixo -

- Suporta métodos de programação funcionais e estruturados, bem como OOP.
- Ele pode ser usado como uma linguagem de script ou pode ser compilado em código de bytes para a construção de grandes aplicativos.
- **Ele fornece tipos de dados** dinâmicos de alto nível e oferece suporte à verificação dinâmica de tipos.
- Suporta coleta automática de lixo.
- Pode ser facilmente integrado com C, C++, COM, ActiveX, CORBA e Java.