

Pitão versus C++

Python é uma linguagem de programação de uso geral e de alto nível. Python é usado para desenvolvimento web, aprendizado de máquina e outros desenvolvimentos de software de ponta. Python é adequado para programadores C++ e Java novos e experientes. **Guido Van Rossam** criou o Python em 1989 no Instituto Nacional de Pesquisa da Holanda. Python foi lançado em 1991.

C++ é uma linguagem de programação orientada a objetos de nível médio, que diferencia maiúsculas de minúsculas. Bjarne Stroustrup criou C++ no Bell Labs. C++ é uma linguagem de programação independente de plataforma que funciona em Windows, Mac OS e Linux. C++ está próximo do hardware, permitindo programação de baixo nível. Isso fornece ao desenvolvedor controle sobre a memória, desempenho aprimorado e software confiável.

Leia este artigo para obter uma visão geral de C++ e Python e como essas duas linguagens de programação são diferentes uma da outra.

O que é Python?

Python é atualmente uma das linguagens de programação mais utilizadas. É uma linguagem de programação interpretada que opera em alto nível. Quando comparado a outras linguagens, a curva de aprendizado do Python é muito menor e também bastante simples de usar.

Python é a linguagem de programação preferida de profissionais que trabalham em áreas como **Inteligência Artificial** , **Aprendizado de Máquina (ML)** , **Ciência de Dados** , **Internet das Coisas (IoT)** , etc., porque se destaca tanto em aplicativos de script quanto como programas autônomos.

Além disso, Python é a linguagem preferida porque é fácil de aprender. Devido à sua excelente sintaxe e legibilidade, a quantidade de dinheiro gasto em manutenção diminui. A modularidade do programa e a capacidade de reutilização do código contribuem para o seu suporte a uma variedade de pacotes e módulos.

Usando Python, podemos realizar -

- desenvolvimento web
- Análise de dados e aprendizado de máquina
- Automação e scripts



- Teste de software e muito mais

Características

Aqui está uma lista de alguns dos recursos importantes do Python -

- **Fácil de aprender** - Python tem uma estrutura simples, poucas palavras-chave e uma sintaxe clara. Isso torna mais fácil para o aluno aprender rapidamente. O código escrito em Python é mais fácil de ler e entender.
- **Fácil de manter** - O código-fonte do Python é muito fácil de acompanhar.
- **Uma grande biblioteca padrão** - A maior parte da biblioteca do Python é fácil de movimentar e funciona em UNIX, Windows, Mac.
- **Portable** - Python pode ser executado em uma ampla variedade de plataformas de hardware e todas elas possuem a mesma interface.

Exemplo de Python

Dê uma olhada no seguinte programa Python simples -

```
a = int(input("Enter value for a"))
b = int(input("Enter value for b"))

print("The number you have entered for a is ", a)
print("The number you have entered for b is ", b)
```

Em nosso exemplo, pegamos duas variáveis "a" e "b" e atribuímos algum valor a essas variáveis. Observe que em Python, não precisamos declarar explicitamente o tipo de dados para variáveis, pois o PVM atribuirá o tipo de dados de acordo com a entrada do usuário.

- A função `input()` é usada para **receber informações do usuário** por meio do teclado.
- Em Python, o tipo de retorno de `input()` é apenas string, então temos que convertê-lo explicitamente para o tipo de dados que precisamos. Em nosso exemplo, convertemos explicitamente para o tipo **int por meio da função `int()`**.
- `print()` é usado para exibir a saída.



Saída

Na execução, este código Python produzirá a seguinte saída -

```
Enter value for a 10
```

```
Enter value for b 20
```

```
The number you have entered for a is 10
```

```
The number you have entered for b is 20
```

O que é C++?

C++ é uma linguagem de programação de uso geral, compilada estaticamente, multiparadigma e com uma curva de aprendizado acentuada. Videogames, aplicativos de desktop e sistemas embarcados o utilizam extensivamente. C++ é tão compatível com C que pode construir praticamente todo o código-fonte C sem quaisquer alterações. A programação orientada a objetos torna C++ uma linguagem melhor estruturada e mais segura que C.

Características

Vamos ver alguns recursos do C++ e o motivo de sua popularidade.

- **Linguagem de nível médio** - É uma linguagem de nível médio, pois pode ser usada tanto para desenvolvimento de sistemas quanto para aplicativos de consumo em larga escala, como Media Players, Photoshop, Game Engines, etc.
- **Velocidade de execução** - O código C++ é executado rapidamente. Porque é compilado e utiliza procedimentos extensivamente. Coleta de lixo, digitação dinâmica e outros recursos modernos impedem a execução do programa.
- **Linguagem orientada a objetos** - A programação orientada a objetos é flexível e gerenciável. Aplicativos grandes são possíveis. O código crescente torna o código processual mais difícil de manusear. A principal vantagem do C++ sobre C.
- **Suporte extensivo à biblioteca** - C++ possui uma vasta biblioteca . Bibliotecas de terceiros são suportadas para desenvolvimento rápido.

Exemplo C++

Vamos entender a sintaxe do C++ através de um exemplo escrito abaixo.

```
#include
using namespace std;

int main() {
    int a, b;
    cout << "Enter The value for variable a \n";
    cin >> a;
    cout << "Enter The value for variable b";
    cin >> b;
    cout << "The value of a is "<< a << "and" << b;
    return 0;
}
```

Em nosso exemplo, estamos recebendo a entrada de duas variáveis "a" e "b" do usuário por meio do teclado e exibindo os dados no console.

Saída

Na execução, produzirá a seguinte **saída** -

```
Enter The value for variable a
10
Enter The value for variable b
20
The value of a is 10 and 20
```

Comparação entre Python e C++ em vários aspectos

Tanto Python quanto C++ estão entre as linguagens de programação mais populares. Ambos têm suas vantagens e desvantagens. Neste tutorial, daremos uma olhada final em seus traços característicos que os diferenciam.

Compilado vs Interpretado

Assim como C, C++ também é uma linguagem baseada em compilador. Um compilador traduz todo o código em um código de linguagem de máquina específico para o sistema operacional em uso e a arquitetura do processador.

Python é uma linguagem baseada em interpretador. O interpretador executa o código-fonte linha por linha.

Plataforma cruzada

Quando um código-fonte C++ como `hello.cpp` é compilado no Linux, ele só pode ser executado em qualquer outro computador com sistema operacional Linux. Se necessário para execução em outro sistema operacional, ele precisará ser compilado.

O interpretador Python não produz código compilado. O código-fonte é convertido em código de bytes sempre que é executado em qualquer sistema operacional, sem quaisquer alterações ou etapas adicionais.

Portabilidade

O código Python é facilmente portátil de um sistema operacional para outro. O código C++ não é portátil, pois deve ser recompilado se o sistema operacional for alterado.

Velocidade de Desenvolvimento

O programa C++ é compilado no código de máquina. Conseqüentemente, sua execução é mais rápida do que a linguagem baseada em intérprete.

O interpretador Python não gera o código de máquina. A conversão do código de bytes intermediário para linguagem de máquina é feita em cada execução do programa.

Se um programa for usado com frequência, C++ é mais eficiente que Python.

Fácil de aprender

Comparado ao C++, o Python possui uma sintaxe mais simples. Seu código é mais legível. Escrever código C++ parece assustador no início por causa de regras de sintaxe complicadas, como o uso de chaves e ponto e vírgula para terminação de frase.

Python não usa colchetes para marcar um bloco de instruções. Em vez disso, ele usa recuos. Declarações de nível de indentação semelhante marcam um bloco. Isso torna um programa Python mais legível.

Digitação estática vs dinâmica

C++ é uma linguagem de tipo estaticamente. O tipo de variáveis para armazenamento de dados precisa ser declarado no início. Variáveis não declaradas não podem ser usadas. Depois que uma variável é declarada como sendo de um determinado tipo, apenas valores desse tipo podem ser armazenados nela.

Python é uma linguagem de tipo dinâmico. Não requer que uma variável seja declarada antes de atribuir-lhe um valor. Como uma variável pode armazenar qualquer tipo de dado, ela é chamada de digitação dinâmica.

Conceitos de OOP

Tanto C++ quanto Python implementam conceitos de programação orientada a objetos. C++ está mais próximo da teoria OOP do que Python. C++ suporta o conceito de encapsulamento de dados, pois a visibilidade das variáveis pode ser definida como pública, privada e protegida.

Python não tem a possibilidade de definir a visibilidade. Ao contrário do C++, o Python não oferece suporte à sobrecarga de métodos. Por ser digitado dinamicamente, todos os métodos são de natureza polimórfica por padrão.

C++ é na verdade uma extensão de C. Pode-se dizer que palavras-chave adicionais são adicionadas em C para que ele suporte OOP. Portanto, podemos escrever um programa orientado a procedimentos do tipo C em C++.

Python é uma linguagem totalmente orientada a objetos. O modelo de dados do Python é tal que, mesmo que você possa adaptar uma abordagem orientada a procedimentos, o Python usa internamente metodologia orientada a objetos.

Coleta de lixo

C++ usa o conceito de ponteiros. A memória não utilizada em um programa C++ não é limpa automaticamente. Em C++, o processo de coleta de lixo é manual. Conseqüentemente, um programa C++ provavelmente enfrentará um comportamento excepcional relacionado à memória.

Python possui um mecanismo de coleta automática de lixo. Conseqüentemente, o programa Python é mais robusto e menos sujeito a problemas relacionados à memória.

Áreas de aplicação

Como o programa C++ é compilado diretamente em código de máquina, ele é mais adequado para programação de sistemas, gravação de drivers de dispositivos, sistemas embarcados e utilitários de sistema operacional.

O programa Python é adequado para programação de aplicativos. Sua principal área de aplicação hoje é ciência de dados, aprendizado de máquina, desenvolvimento de APIs etc.

Diferença entre Python e C++

A tabela a seguir resume as diferenças entre Python e C++ -

Critério	Python	C++
Execução	Python é uma linguagem de programação baseada em interpretação. Os programas Python são interpretados por um intérprete.	C++ é uma linguagem de programação baseada em compilador. Os programas C++ são compilados por um compilador.
Digitando	Python é uma linguagem de tipo dinâmico.	C++ é uma linguagem de tipo estático.
Portabilidade	Python é uma linguagem altamente portátil, o código escrito e executado em um sistema pode ser facilmente executado em outro sistema.	C++ não é uma linguagem portátil, o código escrito e executado em um sistema não pode ser executado em outro sistema sem fazer alterações.
Coleta de lixo	Python fornece um recurso de coleta de lixo. Você não precisa se preocupar com o gerenciamento de memória. É automático em Python.	C++ não fornece coleta de lixo. Você tem que cuidar de liberar memórias. É manual em C++.
Sintaxe	As sintaxes do Python são muito fáceis de ler, escrever e entender.	As sintaxes do C++ são tediosas.
Desempenho	O desempenho de execução do Python é mais lento que o do C++.	Os códigos C++ são mais rápidos que os códigos Python.
Áreas de aplicação	As áreas de aplicação do Python são aprendizado de máquina, aplicações web e muito mais.	As áreas de aplicação do C++ são sistemas embarcados, drivers de dispositivos e muito mais.

