

مقدمه ای بر دستورات پایتون

در این سخنرانی، ما به مرور سریعی بر دستورات پایتون خواهیم پرداخت. این سخنرانی تاکید بر تفاوت های پایتون با زبان های دیگر مانند ++C خواهد داشت.

دو دلیل وجود دارد که ما از این روش برای یادگیری زمینه دستورات پایتون استفاده می کنیم:

1. اگر شما از یک زبان دیگر می آیید، این روش سرعت درک شما از پایتون را به طور قابل توجهی افزایش خواهد داد.
2. یادگیری درباره دستورات به شما اجازه خواهد داد تا در آینده به راحتی زبان های دیگر را بخوانید.

پایتون در مقابل زبان های دیگر

بیا یک دستور ساده ایجاد کنیم که بگوید: "اگر a از b بزرگتر است، 2 را به a و 4 را به b اختصاص دهید"

به این دو دستور if نگاه کنید (ما به زودی درباره ساخت if statements یاد خواهیم گرفت).

نسخه 1 (زبان های دیگر)

```
if (a>b){  
    a = 2;  
    b = 4;  
}
```

نسخه 2 (پایتون)

```
if a>b:  
    a = 2  
    b = 4
```

شما متوجه خواهید شد که پایتون کمتر شلوغ و خواناتر از نسخه اول است. پایتون چگونه این کار را می کند؟

بیا یک از طریق تفاوت های اصلی قدم بزنیم:

پایتون با استفاده از دو عامل اصلی: *کولن* و *فضای سفید*، () و { } را حذف می کند. دستور با یک کولن به پایان می رسد و فضای سفید (تورفتگی) برای توصیف آنچه در صورت بیانیه اتفاق می افتد استفاده می شود.

تفاوت دیگر عمده، عدم وجود نقطه ویرگول در پایتون است. نقطه ویرگول برای نشان دادن پایان دستورات در بسیاری از زبان های دیگر استفاده می شود، اما در پایتون، پایان یک خط همان پایان یک دستور است.

در نهایت، برای پایان دادن به این مرور کوتاه از تفاوت ها، بیا یک به نحوه تورفتگی در پایتون در مقابل زبان های دیگر نگاه دقیق تری داشته باشیم:

تورفتگی

در اینجا کد pseudo-code برای نشان دادن استفاده از فضای سفید و تورفتگی در پایتون وجود دارد:

زبان های دیگر

```
if (x)
    if(y)
        code-statement;
else
    another-code-statement;
```

پایتون

```
if x:
    if y:
        code-statement
else:
    another-code-statement
```

توجه داشته باشید که پایتون به طور قوی توسط تورفتگی کد و فضای سفید محرک می شود. این بدان معنی است که خوانایی کد بخش اصلی طراحی زبان پایتون است.

حالا بیایید با کدنویسی این نوع دستورات در پایتون عمیق تر شویم!

بریم کد بزنیم!