

Python - Argumentos Posicionais

Argumentos posicionais

A **lista** de **variáveis** declaradas entre parênteses no momento da definição de uma **função** são os **argumentos formais** . Uma função pode ser definida com qualquer número de argumentos formais.

Ao chamar uma função -

- Todos os argumentos são obrigatórios
- O número de argumentos reais deve ser igual ao número de argumentos formais.
- **Argumentos formais são posicionais** . Eles pegam valores na ordem de definição.
- O tipo de argumentos deve corresponder.
- Os nomes dos argumentos formais e reais não precisam ser iguais.

Exemplo de argumentos posicionais

```
def add(x,y):  
    z=x+y  
    print ("x={} y={} x+y={}".format(x,y,z))  
  
a=10  
b=20  
add(a,b)
```

Ele produzirá a seguinte **saída** -

```
x=10 y=20 x+y=30
```

Aqui, a função add() possui dois argumentos formais, ambos **numéricos** . Quando os inteiros 10 e 20 passaram para ele. A variável a leva 10 e b leva 20, na ordem de declaração. A função add() exibe a adição.

Python também gera erro quando o número de argumentos não corresponde. Dê apenas um argumento e verifique o resultado.



```
add(b)
```

```
TypeError: add() missing 1 required positional argument: 'y'
```

Passe mais do que um número de argumentos formais e verifique o resultado -

```
add(10, 20, 30)
```

```
TypeError: add() takes 2 positional arguments but 3 were given
```

O tipo de dados dos argumentos reais e formais correspondentes deve corresponder. Altere a para um valor de string e veja o resultado.

```
a="Hello"
```

```
b=20
```

```
add(a,b)
```

Ele produzirá a seguinte **saída** -

```
z=x+y
```

```
~^~
```

```
TypeError: can only concatenate str (not "int") to str
```