

# Blocos de Código

Para entender o funcionamento de blocos de código em Python vamos analisar as motivações do criador da linguagem.

Antes de criar o Python, Guido Van Rossum trabalhou no desenvolvimento de outra linguagem chamada ABC e o objetivo dessa linguagem era ser uma linguagem de fácil leitura por pessoas de outras áreas acadêmicas.

Ele partiu do princípio de que passamos muito mais horas lendo código do que escrevendo e concluiu que a maneira tradicional que as linguagens adotavam para demilitar código não seria tão natural para quem não está acostumado.

Uma grande parte das linguagens utiliza chaves `{ }` para delimitar os blocos de código ficando mais ou menos assim

```
statement (condicao) {  
    primeira linha do bloco;  
    segunda linha do bloco;  
    terceira linha (condicao) {  
        primeira linha do sub bloco;  
    }  
}
```

E para o Guido essa forma de organizar código não seria muito fácil de entender e manter e principalmente não seria tão agradável de ler em códigos grandes.

## Lista de compras

Ao fazer comprar no supermercado geralmente criamos uma lista de compras

```
Feijao  
Sabão  
Arroz  
Batata  
Laranja  
Shampoo  
Alface  
Café
```

Para tornar esta lista mais fácil podemos organizar utilizando as seções do mercado como separador.

```
Mercearia:  
    Feijao  
    Arroz  
    Café  
Limpeza:  
    Sabão  
    Shampoo  
Feira:  
    Batata
```

Laranja  
Alface

A lista acima está muito mais organizado do que a primeira versão e permite que nossa experiência ao fazer compra seja mais produtiva, pois agora podemos percorrer os corredores um a um sem a necessidade de passar duas vezes no mesmo corredor.

## Indentação

Indentation, Indentação ou Denticulação é o termo usado para a formatação da lista de compras acima, após cada categoria ou seção colocamos um **recuo** antes de começar o conteúdo.

E pensando neste exemplo natural o Python foi projetado, de forma que nós passamos muito mais tempo lendo código do que escrevendo.

## Blocos

Em Python um bloco de código inicia sempre que existe a presença de `:` no final de uma linha.

```
if 1 > 2: # inicio de bloco
```

A linha que vem logo após o inicio do bloco deve obrigatoriamente ter um recuo (ou dente) e por isso chamamos de indentação.

```
if 1 > 2:
    # aqui começa o código do bloco
    # o bloco pode ter muitas linhas
    # desde que mantenha o mesmo recuo
    # o recuo padrão é de 4 espaços.
```

Dentro de um bloco de código podem existir muitos sub blocos, níveis internos de recuo, mas a recomendação é que no máximo existam 4.

```
if 1 > 2:
    # aqui inicia o bloco
    # recuo de 4 espaços

    while x < 10:
        # aqui inicia outro sub bloco
        # recuo de 8 espaços

        if x == 3:
            # ainda mais um bloco
            # recuo de 12 espaços

        # voltamos ao bloco anterior

    # agora voltamos para o bloco inicial

# e aqui continuamos o bloco principal (main)
```

A maior parte dos editores de código possui ferramentas que ajudam a visualizar as linhas de indentação.