

Le Langage Go

Les bases du langage

Boucle for

**Faire des itérations tant qu'une
condition est respectée**

for flexible en Go

Syntaxe (longue)

Syntaxe (longue)

```
for <initialisation>; <condition>; <action de boucle> {  
    // code  
}
```

Example

```
for i := 0; i < 5; i++
```

Example

```
for i := 0; i < 5; i++ {  
    fmt.Println(i) //output, 0, 1, 2, 3, 4  
}
```

Syntaxe "while"

Syntaxe "while"

```
for <condition> {  
    // code  
}
```

Syntaxe "while"

```
for <condition> {  
    // code  
}
```

```
i := 1  
for i <= 3 {  
    fmt.Println(i) // output: 1, 2, 3  
    i = i + 1  
}
```

Syntaxe "forever"

Syntaxe "forever"

```
for {  
    fmt.Println("Loooop") // output: Loooop, Loooop, Loooop, ...  
}
```