

# Range

C'est la continuité du `for`, il permet d'itérer sur une collection de donnée

## Syntaxe

```
for <index>, <value> := <dataset> {  
    //code  
}
```

## Exemple

```
names := []string{"Bob", "Alice", "Bobette", "John"}  
for i, n := range names {  
    fmt.Printf("Username=%s (index=%d)\n", n, i)  
}  
  
// range on string  
// Omit index / value  
for _, c := range "golang" {  
    fmt.Printf("%v\n", string(c))  
}
```

---

Revision #1

Created 9 May 2022 19:10:50 by Noé Larrieu-Lacoste

Updated 9 May 2022 19:11:20 by Noé Larrieu-Lacoste