

# Циклы шпаргалка

## Циклы в Go

Циклы в Go бывают только вида `for`. Цикл состоит из трёх компонентов: инициализация переменной, условие, шаг.

```
for инициализация; условие; шаг {  
    // выполняемые действия  
}
```

Цикл продолжается до тех пор, пока условие не станет ложным:

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

## Цикл с проверкой условия

Цикл `for` мы можем использовать как `while`. Цикл будет выполняться до тех пор, пока условие не станет истинным.

```
i := 0  
for i < 5 {  
    fmt.Println(i)  
    i++ // увеличиваем итератор  
}
```

В этом примере цикл будет повторён до тех пор, пока `i` меньше 5. Переменная `i` также увеличивается на 1 после каждой итерации.

## Break, Continue

Оператор `break` используется для выхода из цикла до его завершения, в то время как оператор `continue` используется для пропуска текущей итерации цикла и перехода к следующей. Пример использования `break` и `continue`:

```
for i := 0; i < 10; i++ {
    if i == 3 {
        continue
    }
    if i == 8 {
        break
    }
    fmt.Println(i)
}
```

## Бесконечные циклы

Бесконечный цикл — тот, который продолжается бесконечно.

```
for {
    fmt.Println("This is an infinite loop")
}
```

Прервать бесконечный цикл можно с помощью оператора `break`.

```
func main() {
    for {
        var input string
        fmt.Scanln(&input)
        if input == "exit" {
            break // прерываем бесконечный цикл
        }
    }
}
```