

# Kotlin Cheat Sheet: Control Structures

---

*Java*

```
if (score >= 0 && score <= 300) { }
```

*Kotlin*

```
if (score in 0..300) { }
```

*Java*

```
if (score >= 0 && score <= 300) { }
```

*Kotlin*

```
if (score in 0..300) { }
```

```
int score = // some score;
String grade;
switch (score) {
    case 10:
    case 9:
        grade = "Excellent";
        break;
    case 8:
    case 7:
    case 6:
        grade = "Good";
        break;
    case 5:
    case 4:
        grade = "OK";
        break;
    case 3:
    case 2:
    case 1:
        grade = "Fail";
        break;
    default:
        grade = "Fail";
}
```

## Kotlin

```
var score = // some score
var grade = when (score) {
    9, 10 -> "Excellent"
    in 6..8 -> "Good"
    4, 5 -> "OK"
    in 1..3 -> "Fail"
    else -> "Fail"
}
```

## Java

```
int score = // some score;
String grade;
switch (score) {
    case 10:
    case 9:
        grade = "Excellent";
        break;
    case 8:
    case 7:
    case 6:
        grade = "Good";
        break;
    case 5:
    case 4:
        grade = "OK";
        break;
    case 3:
    case 2:
    case 1:
        grade = "Fail";
        break;
    default:
        grade = "Fail";
}
```

## Kotlin

```
var score = // some score
var grade = when (score) {
    9, 10 -> "Excellent"
    in 6..8 -> "Good"
    4, 5 -> "OK"
    in 1..3 -> "Fail"
    else -> "Fail"
}
```

## Java

```
for (int i = 1; i <= 10 ; i++) { }
```

```
for (int i = 1; i < 10 ; i++) { }
```

```
for (int i = 10; i >= 0 ; i--) { }
```

```
for (int i = 1; i <= 10 ; i+=2) { }
```

```
for (int i = 10; i >= 0 ; i-=2) { }
```

```
for (String item : collection) { }
```

```
for (Map.Entry<String, String> entry: map.entrySet()) { }
```



## *Kotlin*

```
for (i in 1..10) { }  
  
for (i in 1 until 10) { }  
  
for (i in 10 downTo 0) { }  
  
for (i in 1..10 step 2) { }  
  
for (i in 10 downTo 1 step 2) { }  
  
for (item in collection) { }  
  
for ((key, value) in map) { }
```

## Java

```
for (int i = 1; i <= 10 ; i++) { }  
  
for (int i = 1; i < 10 ; i++) { }  
  
for (int i = 10; i >= 0 ; i--) { }  
  
for (int i = 1; i <= 10 ; i+=2) { }  
  
for (int i = 10; i >= 0 ; i-=2) { }  
  
for (String item : collection) { }  
  
for (Map.Entry<String, String> entry: map.entrySet()) { }
```

## Kotlin

```
for (i in 1..10) { }  
  
for (i in 1 until 10) { }  
  
for (i in 10 downTo 0) { }  
  
for (i in 1..10 step 2) { }  
  
for (i in 10 downTo 1 step 2) { }  
  
for (item in collection) { }  
  
for ((key, value) in map) { }
```