

5. Strings

A string is an ordered collection of characters, such as "We ♥ Swift" or "eat, sleep, code, repeat!". In Swift strings are represented by the `String` type which is a collection of values of `Character` type.

Creating strings

You can include predefined `String` values within your code as string literals. A string literal is a fixed sequence of textual characters surrounded by a pair of double quotes (`"`).

```
let aString = "Hello"
```

Take note of a special case of string, the empty string.

```
var emptyString = "" // empty string literal
```

You can create a new string by concatenating two strings using the `+` operator.

```
let newString = "Hello" + " swift lovers" // "Hello swift lovers"
```

String interpolation can also be used to combine strings. By using the `\(<value>)` syntax inside a string literal.

```
let bat = "BAT"
let man = "MAN"

// "BATMAN" - \(<bat>) will be replaced with "BAT" and \(<man>) with "MAN"
let batman = "\(<bat>)\(<man>)"

print("\(<bat>) + \(<man>) = \(<batman>)")
// "BAT + MAN = BATMAN"
```

Characters

In some cases you will want to work with the individual characters that make up the string. To do that you can use the **for-in** syntax. A string exposes the list of characters with the `characters` property.

```
var someString = "this string has 29 characters"
```

```
for character in someString.characters {  
    // to convert a Character into a String use string interpolation  
    // you will need to do this in order to compare characters  
    var characterAsString = "\(character)"  
  
    print(characterAsString)  
}
```

To see how many character a string has we can use the `count` property on the `characters` property of the string.

```
var string = "This string has 29 characters"  
print(string.characters.count) // 29
```

Comparing

You can compare string using the same operators as numbers, but usually you only care about equality.

```
"We ♥ Swift" == "We ♥ Swift" // true  
"We ♥ Swift" == "we ♥ swift" // false - string comparison is case sensitive  
  
"We ♥ Swift" != "We ♥ Swift" // false  
"We ♥ Swift" != "we ♥ swift" // true
```