Kotlin Operators

Operators are special symbols (characters) that carry out operations on operands (variables and values). For example, + is an operator that performs addition.

Arithmetic Operators

Operator	Meaning/ Used for
+	Addition, String concatenation
-	Subtraction
*	Multiplication
/	Division
%	Modulus, used to find remainder of division operation

Examples:

Addition

```
fun main(){
  var num1 = 50
  var num2 = 30
  var sum = num1 + num2
  print(sum)
}
```

Subtraction

```
fun main(){
  var num1 = 45
  var num2 = 21
  var difference = num1 - num2
  print(difference)
}
```

Multiplication

```
fun main(){
  var num1 = 650
  var num2 = 11
  var product = num1 * num2
  print(product)
}
```

Division

```
fun main(){
  var num1 = 1950
  var num2 = 32
  var result = num1 / num2
  print(result)
}
```

Modulus

```
fun main(){
  var num1 = 240
  var num2 = 45
  var remainder = num1 % num2
  print(remainder)
}
```

Assignment Operators

The assignment operator is the equal sign =. We use it to assign values to variables e.g

```
var phone = "Samsung"
var price = 10000
```

We can also combine the equal sign with other operators to shorten other operations like:

Expression	Equivalent Operation	Explanation
a += b	a = a + b	Add b to the current value of a then assign the result back to a
a -= b	a = a - b	Subtract b from the current value of a then assign the result back to a
a *= b	a = a * b	Multiply the current value of a by b then assign the result back to a
a /= b	a = a / b	Divide the current value of aby b then assign the result back to a
a %= b	a = a % b	Divide the current value of a by b then assign the remainder to a

Increment, Decrement, Comparison and Equality Operators

Operator	Use / Meaning	Explanation / Example	Result

++	Increment: Increase value by 1	`fun main(){`
		`  var age = 19`
		`  ++age`
		`  print(age)`
		`}`
			`20`

--	Decrement: Decrease value by 1	`fun main(){`
		`  var weight = 52`
		`  --weight`
		`  print(weight)`
		`}`
			`51`

>	Greater than	`fun main(){`
		`  val a = 325`
		`  val b = 432`
		`  print(a>b)`
		`}`
			`false`

<	Less than	`fun main(){`
		`  val a = 325`
		`  val b = 432`
		`  print(a<b)`
		`}`
			`true`

>=	Greater than or equal to	`fun main(){`
		`  val a = 325`
		`  val b = 432`
		`  print(a>=b)`
		`}`
			`false`

<=	Less than or equal to	`fun main(){ val a = 325 val b = 432 print(a<=b) }`	`true`
==	is equal to	`fun main(){ val a = 325 val b = 432 print(a==b) }`	`false`
!=	not equal to	`fun main(){ val a = 325 val b = 432 print(a!=b) }`	`true`

Logical Operators

To be discussed together with flow control.