Arithmetic Operators

In the following lesson, you will be introduced to arithmetic operators.

Arithmetic operators are operators that perform arithmetic operations such as addition and subtraction. Below is a list of the arithmetic operators supported by Scala.

Operator	Use
`+`	Adds two operands
`-`	Subtracts the second operand from the first
`*`	Multiplies both operands
`/`	Divides the first operand by the second operand
`%`	Finds the remainder after division of one number by another

Taking the first operand to be 10 and the second operand to be 7, let's look at an example for each operator.

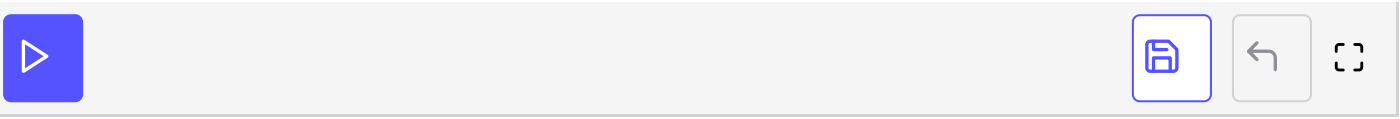
This code requires the following environment variables to execute:

LANG	C.UTF-8

```
val operand1 = 10
val operand2 = 7
```

```
println(operand1 + operand2)
println(operand1 - operand2)
println(operand1 * operand2)

println(operand1 / operand2)
println(operand1 % operand2)
```



The output we see when we press run is the expected output. However, if you input `10/7` on your calculator, you surely won't get 1. So why did Scala give us 1?

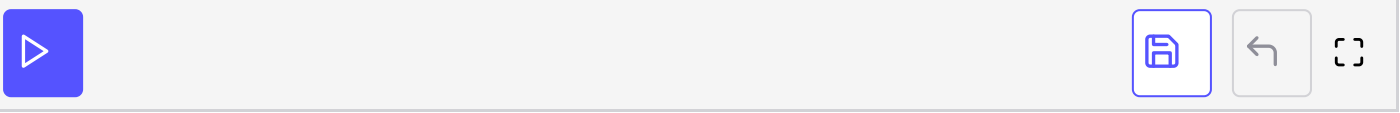
The reason for this is that our operands are of type `Int`, hence our output will also be of type `Int`. To keep the types consistent, Scala will only tell you the whole number part of the quotient, excluding any remainder. If you also want the remainder, all you have to do is specify, when declaring your variables, that the operands are either of type `Float` or `Double`, depending on your requirements.

This code requires the following environment variables to execute:

LANG	C.UTF-8

```
val operand1 = 10F
val operand2 = 7F

println(operand1 + operand2)
println(operand1 - operand2)
println(operand1 * operand2)
println(operand1 / operand2)
println(operand1 % operand2)
```



Running the above code should now give you an output of type `Float`.

That sums up arithmetic operators, let's move on to relational operators in the next lesson.