

Type Casting in Scala

In the following lesson, you will be introduced to type casting and learn how to change the data type of a variable.

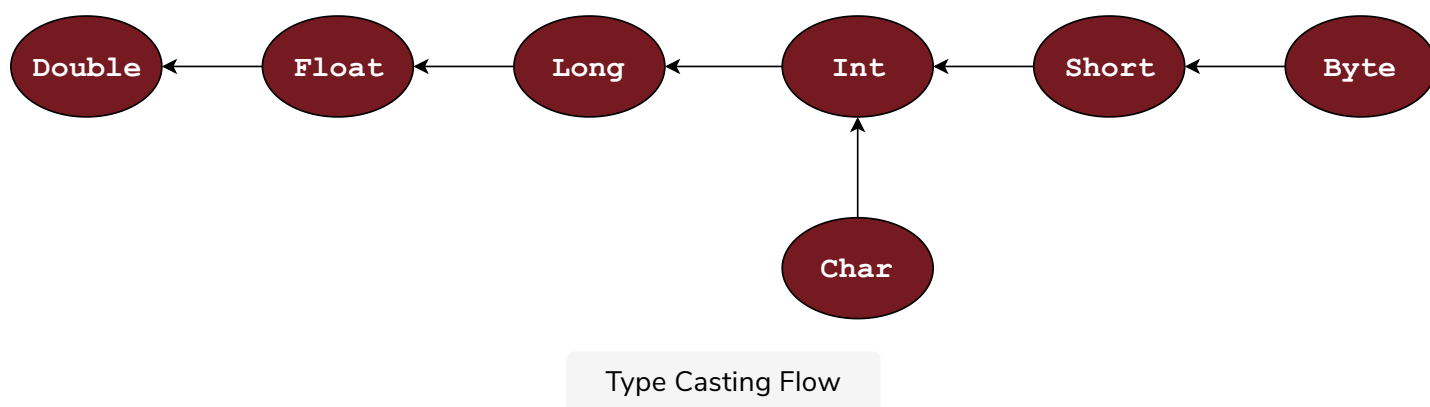
We'll cover the following ^

- Introduction
- Long => Float
- Char => Int

Introduction

Sometimes we find ourselves in a situation where we need the value of an already existing variable in a different data type than the one with which it was defined. This is where type casting comes in. Type casting is an approach which allows changing the data type of a variable or an expression from one to another.

One thing to remember is that not every data type can be converted to a data type of choice. Let's look at the type casting flow in Scala to better understand which data type can be converted to which data type.



The arrows in the above diagram are giving the direction of conversion. For example, a `Float` can be converted to a `Double`, but a `Double` cannot be converted to a `Float`. The elements more or less remain the same and the value is simply altered with respect to the new data type.

Let's look at some examples.

Long => Float

In the code below, the variable `oldType` is of type `Long`. However, we want the value of `oldType` to be of type `Float`. What we will do is create a new variable `newType` of type `Float` and assign it the value of `oldType`. Type casting will ensure that the value is altered according to the data type. Run the program below and see what happens.

This code requires the following environment variables to execute:

LANG C.UTF-8

```
val oldType: Long = 926371285
val newType: Float = oldType

// Driver Code
println(oldType)
println(newType)
```



While the above program successfully compiles, the code below won't because we are trying to convert a `Float` to a `Long`.

This code requires the following environment variables to execute:

LANG C.UTF-8

```
val oldType: Long = 926371285
val newType: Float = oldType
val newOldType: Long = newType

// Driver Code
println(oldType)
println(newType)
println(newOldType)
```



After pressing **RUN** you should see a runtime error: type mismatch. This is because `newType` is of type `Float` and on **line 3** we are trying to type cast it to `Long`. From our type cast flow chart, we know that this is not allowed.

Char => Int

The code below converts the `Char` 'A' to an `Int`

The code below converts the `Char` `A` to an `Int`.

This code requires the following environment variables to execute:

LANG

C.UTF-8

```
val oldType: Char = 'A'
val newType: Int = oldType

// Driver Code
println(oldType)
println(newType)
```



Every character has a Decimal [ASCII](#) value. In the example above, because there is no integer `A`, the character `A` will be converted to its ASCII value, i.e. `65`.

In the next lesson, you will be challenged to type cast a variable.