

# Scalacheck

Scalacheck is a library for property based testing. Property based testing libraries are very useful to detect corner cases in our domain logic.

we use it with the import:

```
scala> import org.scalacheck._  
import org.scalacheck._
```

# Properties

Properties are the name we give to assertions in this kind of testing. The difference with assertions is that instead of testing them with just one value, we will pass a bunch of randomly generated values.

## Generating values

In order to generate values for a type, we need a generator (called `Gen` in Scalacheck).

Basic generators (**see in the console the following**):

- ▶ `Gen.alphaUpperStr`
- ▶ `Gen.posNum[Int]`
- ▶ `Gen.posNum[Double]`
- ▶ `Gen.negNum[Int]`
- ▶ `Gen.oneOf`
- ▶ `Gen.listOf`
- ▶ `Gen.const`

We can construct more interesting generators using `Gen`'s functional combinators.

```
scala> case class Car(age: Int, brand: String)
```