**Class 6 : closure**

A closure in Groovy is an open, anonymous, block of code that can take arguments, return a value and be assigned to a variable. A closure may reference variables declared in its surrounding scope. In opposition to the formal definition of a closure, Closure in the Groovy language can also contain free variables which are defined outside of its surrounding scope. While breaking the formal concept of a closure, it offers a variety of advantages which are described in this chapter.

A closure is just like a method except it is a first class citizen of the language. When I have a method in a class we give it a name. It takes some arguments and performs some actions. A closure will do the same thing. But unlike that method a closure is an object and can be used or passed around your program's.

**What are Closures used for –**

* Iterators
* Callbacks
* Higher-order functions
* Specialized Control Structure
* Builders
* Resource allocation
* Threads
* DSLs
* Fluent Interfaces

**Delegates :**

* When append is called using writer closure , then it looks outside of itself into the enclosing class.
* In our case it is script, so writer closer looks in enclosing class that if you have append method
* Then it looks for what is the delegate for this particular closure.