***Handling text***

* Just as in Java, character data is mostly handled using the java.lang.String class. But Groovy provides some tweaks to make that easier, with more options for string literals and some helpful operators.

**GSTRINGS**

In Groovy, string literals can appear in single or double quotes. The double-quoted version allows the use of placeholders, which are automatically resolved as required. This is a *GString*, and that’s also the name of the class involved. The following code demonstrates a simple variable expansion, although that’s not all GStrings can do:

def nick = 'ReGina'

def book = 'Groovy in Action, 2nd ed.'

assert "$nick is $book" == 'ReGina is Groovy in Action, 2nd ed.'

***Numbers are objects***

* Groovy *numbers* have a familiar appearance, but unlike in Java, they’re first-class objects rather than primitive types.
* In Java, you cannot invoke methods on primitive types. If x is of primitive type int, you cannot write x.toString().
* Int Groovy you can use numbers with numeric operators, and you can also call methods on number instances. For example:

def x = 1

def y = 2

assert x + y == 3

assert x.plus(y) == 3

assert x instanceof Integer

* The variables x and y are objects of type java.lang.Integer. Thus, you can use the plus method, but you can just as easily use the + operator.