**String in Groovy**

A string is a sequence of characters. Generally, a string is a data type and implemented as an array of bytes which stores sequence of characters. In Groovy, String class is available in java.lang.String objects as well as in groovy.lang. GString which is called interpolated strings in some programming language.

*Note: In groovy, all the strings can be concatenated by using the + operator*

Single-quoted string

In groovy, Single-quoted strings is a sequence of characters which is enclosed under single quotes.

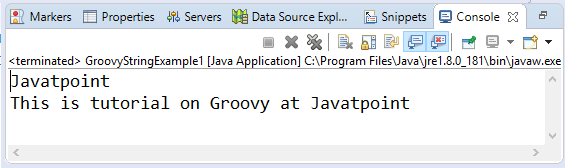
Syntax:

1. '\_\_\_\_\_\_\_\_String\_\_\_\_\_\_\_\_'

Example 1:

1. **package** com.app
2. **class** GroovyStringExample1 {
3. **static** **void** main(args)
4. {
5. String s1 = 'Javatpoint'
6. println s1
7. println 'This is tutorial on Groovy at ' + s1
8. }
9. }

**Output:**



Double-quoted string

In groovy, Double-quoted string is a sequence of characters which is enclosed under double quotes.

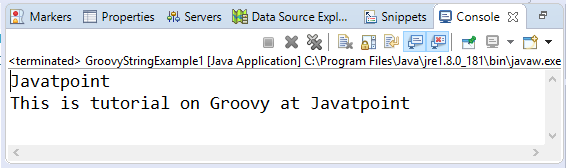
Syntax:

1. "\_\_\_\_\_\_\_\_String\_\_\_\_\_\_\_\_"

Example 2:

1. **package** com.app
2. **class** GroovyStringExample2 {
3. **static** **void** main(args)
4. {
5. String s1 = "Javatpoint"
6. println s1
7. println "This is tutorial on Groovy at " + s1
8. }
9. }

**Output:**

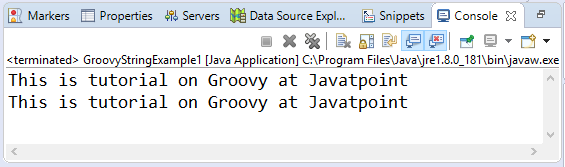


In Groovy, we can also use **${Variable \_name}** and **$Variable\_name** instead of using the '+' operator. Using **$Variable\_name** is known as interpolation. It can be used only when the string is defined inside the double quotes.

Example 3:

1. **package** com.app
2. **class** GroovyStringExample3 {
3. **static** **void** main(args)
4. {
5. String s1 = "Javatpoint"
6. println "This is tutorial on Groovy at ${s1} "
7. println "This is tutorial on Groovy at $s1 "
8. }
9. }

**Output:**



Triple-single-quoted string

In groovy, Triple-single-quoted string is a sequence of characters which is enclosed under triple single quotes.

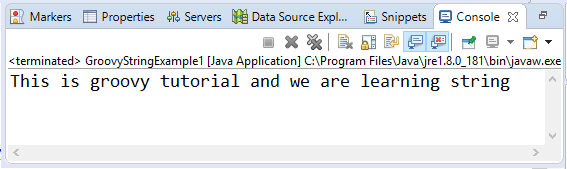
Syntax:

1. '''\_\_\_\_\_\_\_\_String\_\_\_\_\_\_\_\_'''

Example 4:

1. **package** com.app
2. **class** GroovyStringExample4 {
3. **static** **void** main(args)
4. {
5. String s1 = '''This is groovy tutorial and we are learning string'''
6. println s1
7. }
8. }

**Output:**

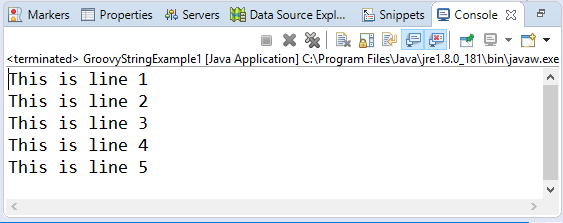


In Groovy, Triple-single-quoted strings are mostly used for multiple lines in a single variable.

Example 5:

1. **package** com.app
2. **class** GroovyStringExample5 {
3. **static** **void** main(args)
4. {
5. String s1 = '''This is line 1
6. This is line 2
7. This is line 3
8. This is line 4
9. This is line 5'''
10. println s1
11. }}

**Output:**



Triple-double-quoted string

In groovy, Triple-double-quoted string is a sequence of characters which is enclosed under triple double quotes.

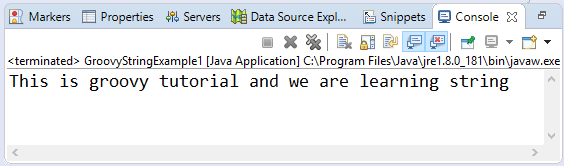
Syntax:

1. """\_\_\_\_\_\_\_\_String\_\_\_\_\_\_\_\_"""

Example 6:

1. **package** com.app
2. **class** GroovyStringExample6 {
3. **static** **void** main(args)
4. {
5. String s1 = """This is groovy tutorial and we are learning string"""
6. println s1
7. }
8. }

**Output:**

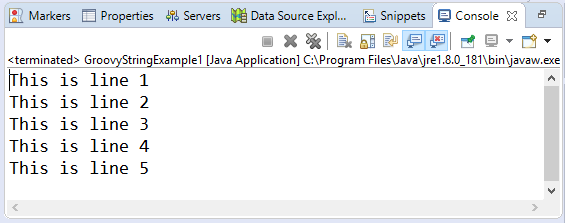


In Groovy, Triple-double-quoted strings are also used for multiple lines in a single variable.

Example 7:

1. **package** com.app
2. **class** GroovyStringExample7 {
3. **static** **void** main(args)
4. {
5. String s1 = """This is line 1
6. This is line 2
7. This is line 3
8. This is line 4
9. This is line 5"""
10. println s1
11. }
12. }

**Output:**

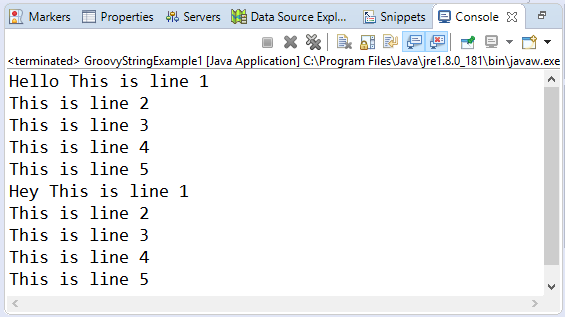


In Triple-double-quoted strings, we can also use **${Variable \_name}** and **$Variable\_name** instead of using the '+' operator. Using **$Variable\_name** is known as interpolation.

Example 8:

1. **package** com.app
2. **class** GroovyStringExample8 {
3. **static** **void** main(args)
4. {
5. String s1 = """This is line 1
6. This is line 2
7. This is line 3
8. This is line 4
9. This is line 5"""
10. println """Hello $s1"""
11. println """Hey $s1"""
12. }
13. }

**Output:**



Slashy string

In groovy, slashy string is a sequence of characters which is enclosed in / forward slash. Slashy strings are useful for defining regular expressions and patterns where there is no need to escape backslashes.

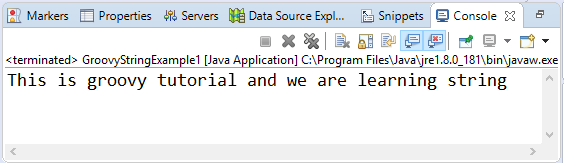
Syntax:

1. /\_\_\_\_\_\_\_\_String\_\_\_\_\_\_\_\_/

Example 9:

1. **package** com.app
2. **class** GroovyStringExample9 {
3. **static** **void** main(args)
4. {
5. String s1 = /This is groovy tutorial and we are learning string/
6. println s1
7. }
8. }

**Output:**

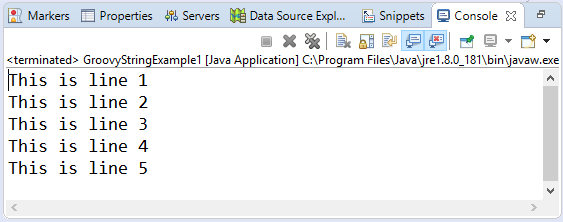


In Groovy, slashy strings are also used for multiple lines in a single variable.

Example 10:

1. **package** com.app
2. **class** GroovyStringExample11 {
3. **static** **void** main(args)
4. {
5. String s1 = /This is line 1
6. This is line 2
7. This is line 3
8. This is line 4
9. This is line 5/
10. println s1
11. }}

**Output:**

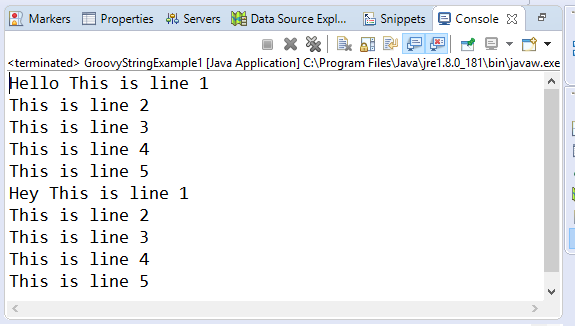


In slashy strings we can also use **${Variable \_name}** and **$Variable\_name** instead of using the '+' operator. Using **$Variable\_name** is known as interpolation.

Example 11:

1. **package** com.app
2. **class** GroovyStringExample11 {
3. **static** **void** main(args)
4. {
5. String s1 = /This is line 1
6. This is line 2
7. This is line 3
8. This is line 4
9. This is line 5/
10. println """Hello ${s1}"""
11. println """Hey $s1"""
12. }}

**Output:**



Dollar slashy string

In groovy, dollar slashy string is a sequence of characters which is enclosed in the $/ dollar and a forward slash. Dollar slashy strings are useful for defining regular expressions and patterns where there is no need to escape backslashes.

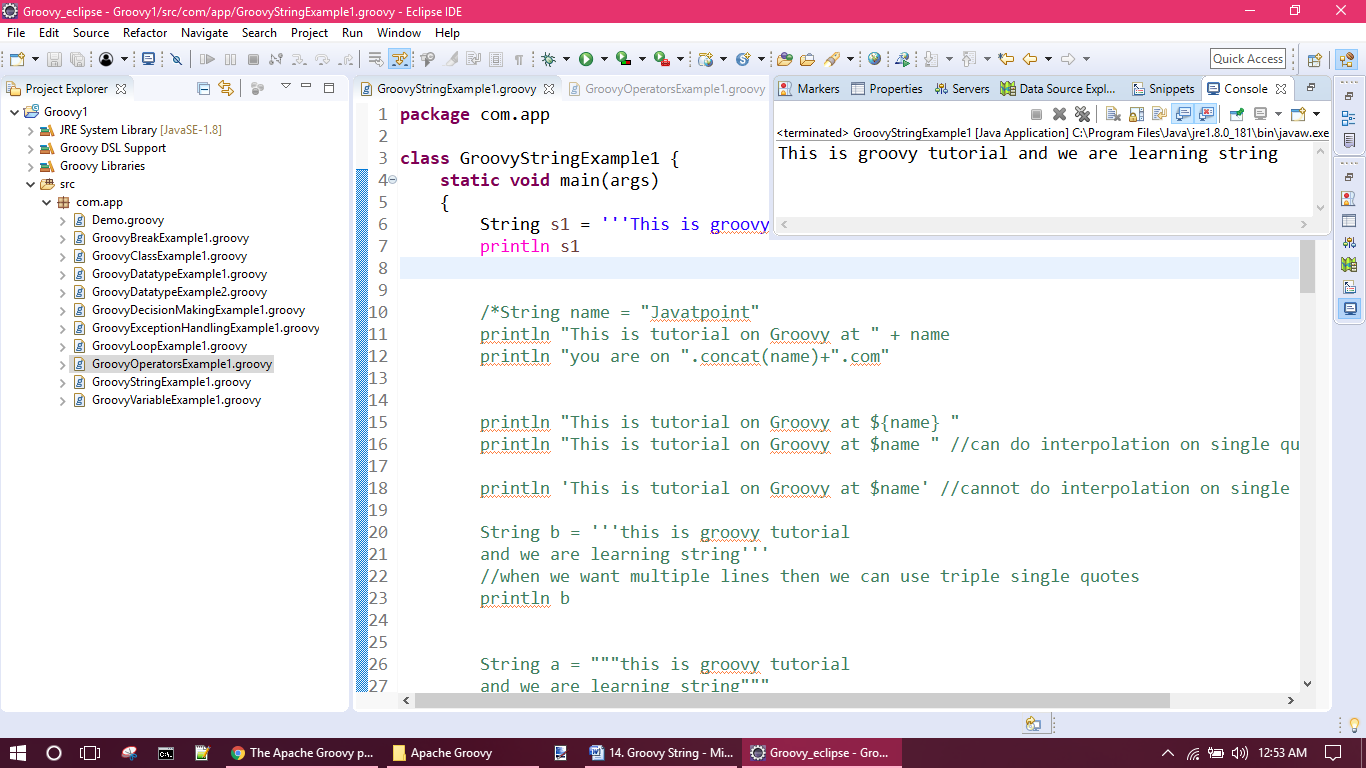
Syntax:

1. $/\_\_\_\_\_\_\_\_String\_\_\_\_\_\_\_\_/$

Example 12:

1. **package** com.app
2. **class** GroovyStringExample9 {
3. **static** **void** main(args)
4. {
5. String s1 = $/This is groovy tutorial and we are learning string /$
6. println s1
7. }
8. }

**Output:**

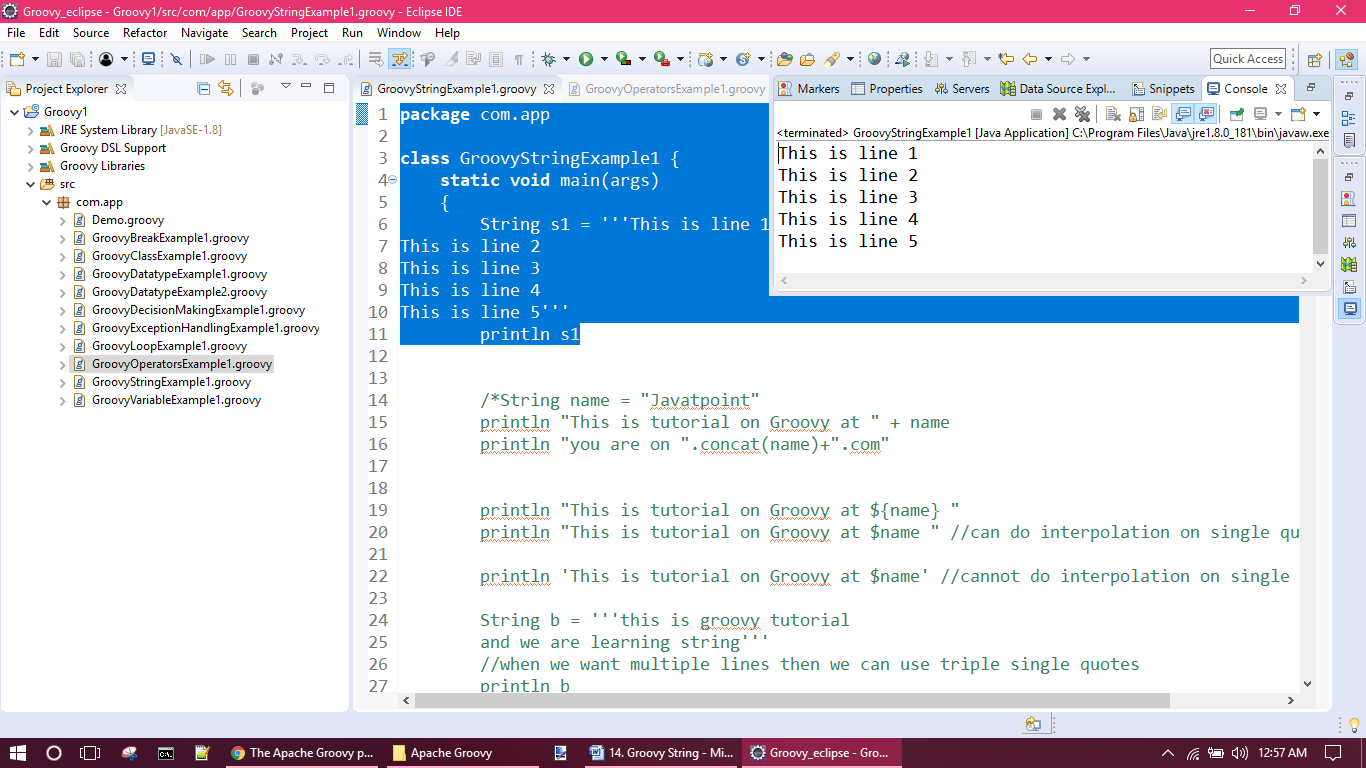


In Groovy, dollar slashy strings are also used for multiple lines in a single variable.

Example 13:

1. **package** com.app
2. **class** GroovyStringExample13 {
3. **static** **void** main(args)
4. {
5. String s1 = $/This is line 1
6. This is line 2
7. This is line 3
8. This is line 4
9. This is line 5/$
10. println s1
11. }
12. }

**Output:**



In dollar slashy strings, we can also use **${Variable \_name}** and **$Variable\_name** instead of using the '+' operator. Using **$Variable\_name** is known as interpolation.

Example 11:

1. **package** com.app
2. **class** GroovyStringExample11 {
3. **static** **void** main(args)
4. {
5. String s1 = $/This is line 1
6. This is line 2
7. This is line 3
8. This is line 4
9. This is line 5/$
10. println """Hello ${s1}"""
11. println """Hey $s1"""
12. }}

**Output:**

