Groovy lab exercise-2

**Exercises**:

* Create a set from a list containing duplicate elements. What do you observe? How can you achieve the same result without converting a list to a set?

Sol:-

List l=[1,1,1,2,2,3,4,5,6]

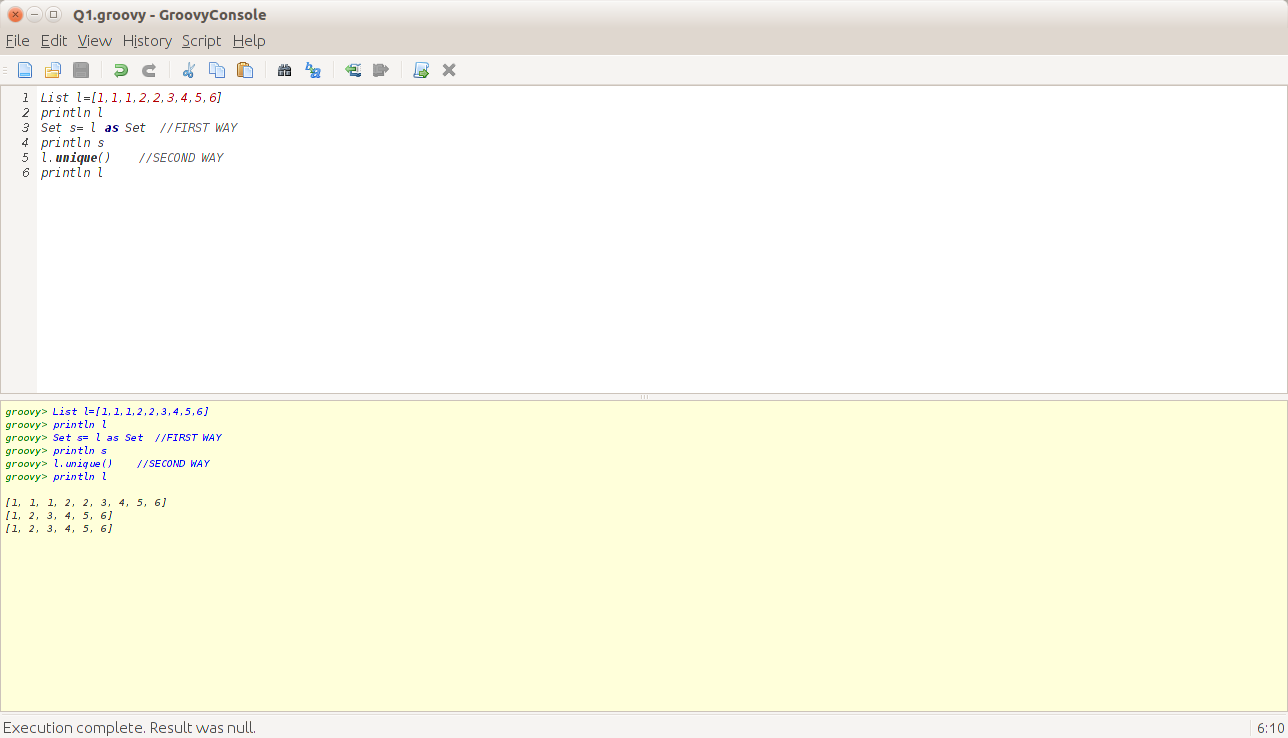
println l

Set s= l as Set //FIRST WAY

println s

l.unique() //SECOND WAY

println l



* Get first, second and last element of Range.

Sol:

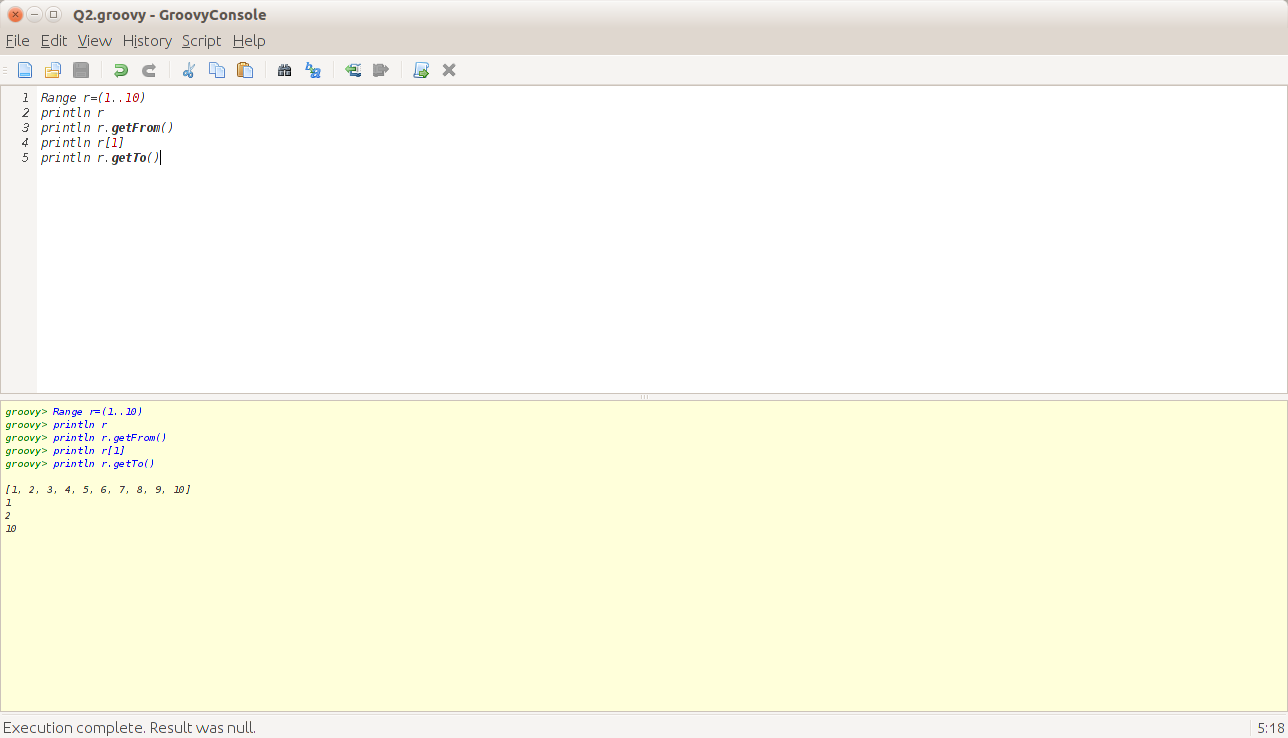
Range r=(1..10)

println r

println r.getFrom()

println r[1]

println r.getTo()



* Print the table of a given number : 2 and 12

Sol:-

(1..10).each{

println "2\*${it} =${2\*it}";

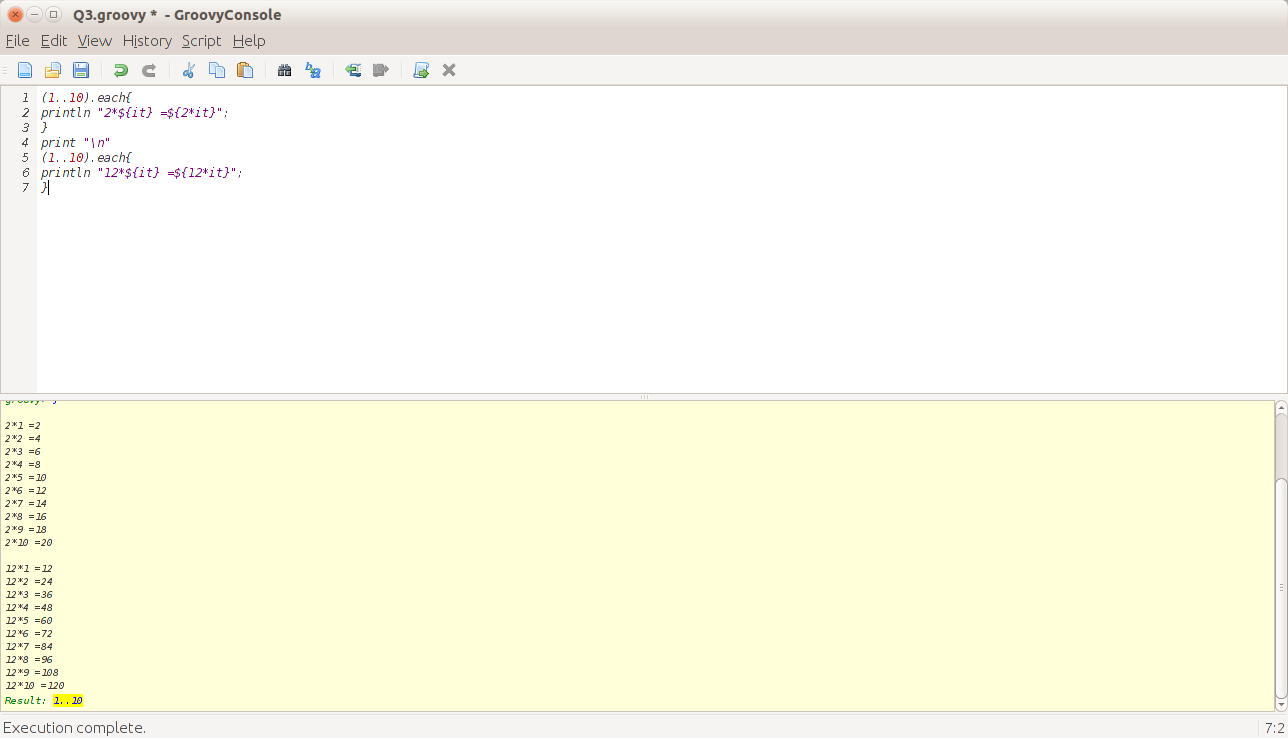
}

print "\n"

(1..10).each{

println "12\*${it} =${12\*it}";

}



* We have a sorted list of alphabets a-z, print all alphabets appearing after "j"

Sol:-

List l=('a'..'z')

println l

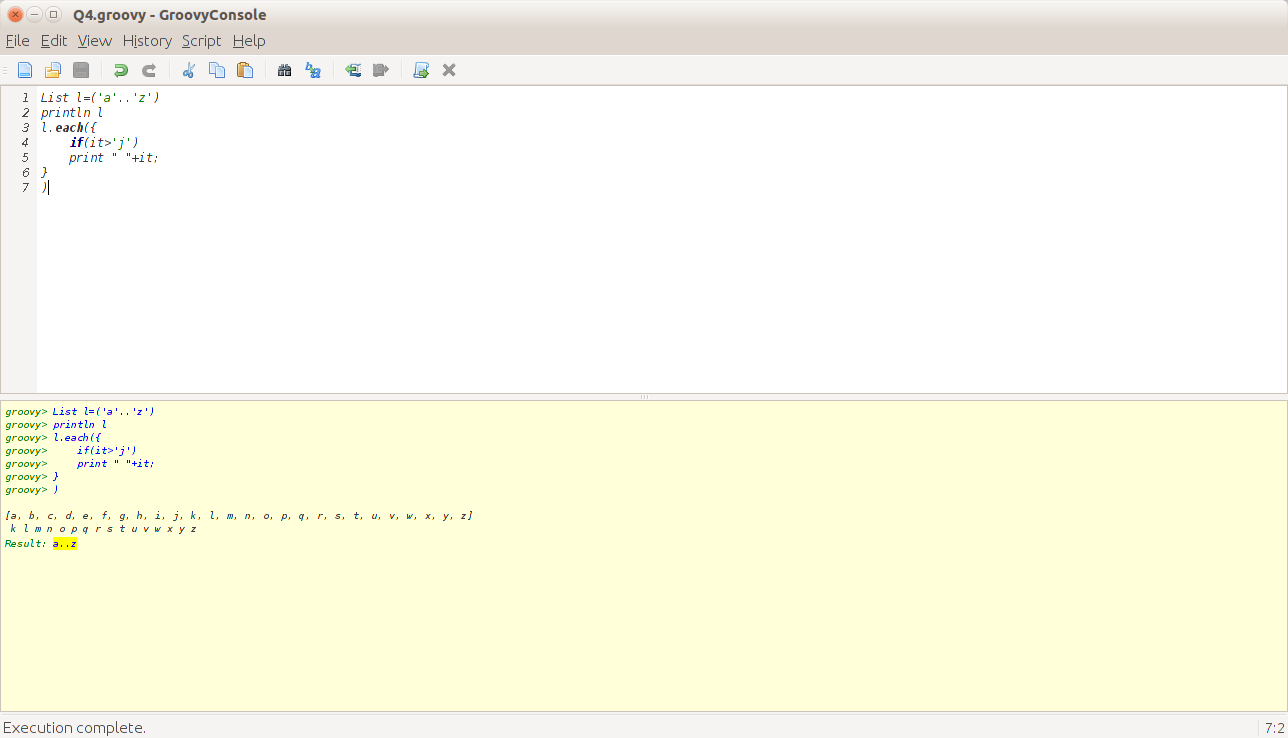
l.each({

if(it>'j')

print " "+it;

}

)

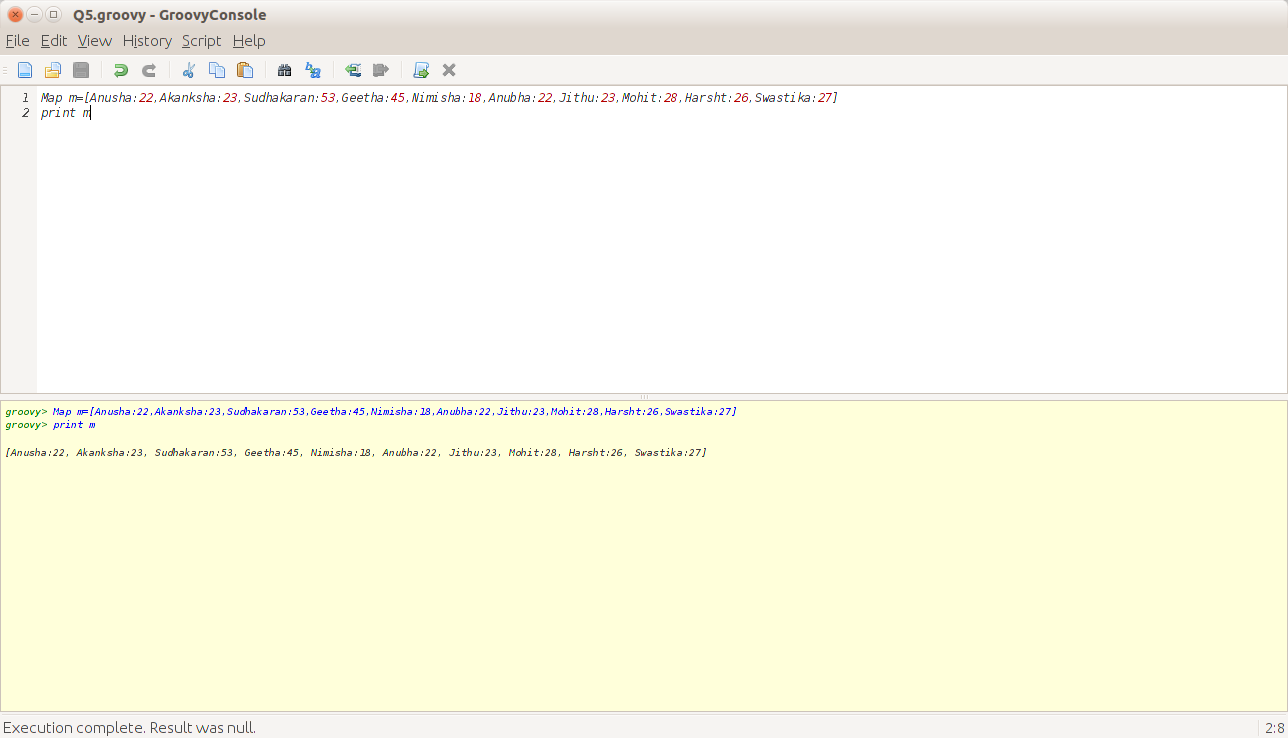


* Create a new map consisting of 10 of your friend's name's as keys and their ages as value.

Sol:-

Map m=[Anusha:22,Akanksha:23,Sudhakaran:53,Geetha:45,Nimisha:18,Anubha:22,Jithu:23,Mohit:28,Harsht:26,Swastika:27]

print m



* Iterate over the previous map in as many ways as possible

Sol:-

Map m=[Anusha:22,Akanksha:23,Sudhakaran:53,Geetha:45,Nimisha:18,Anubha:22,Jithu:23,Mohit:28,Harsht:26,Swastika:27]

print m.each{

it

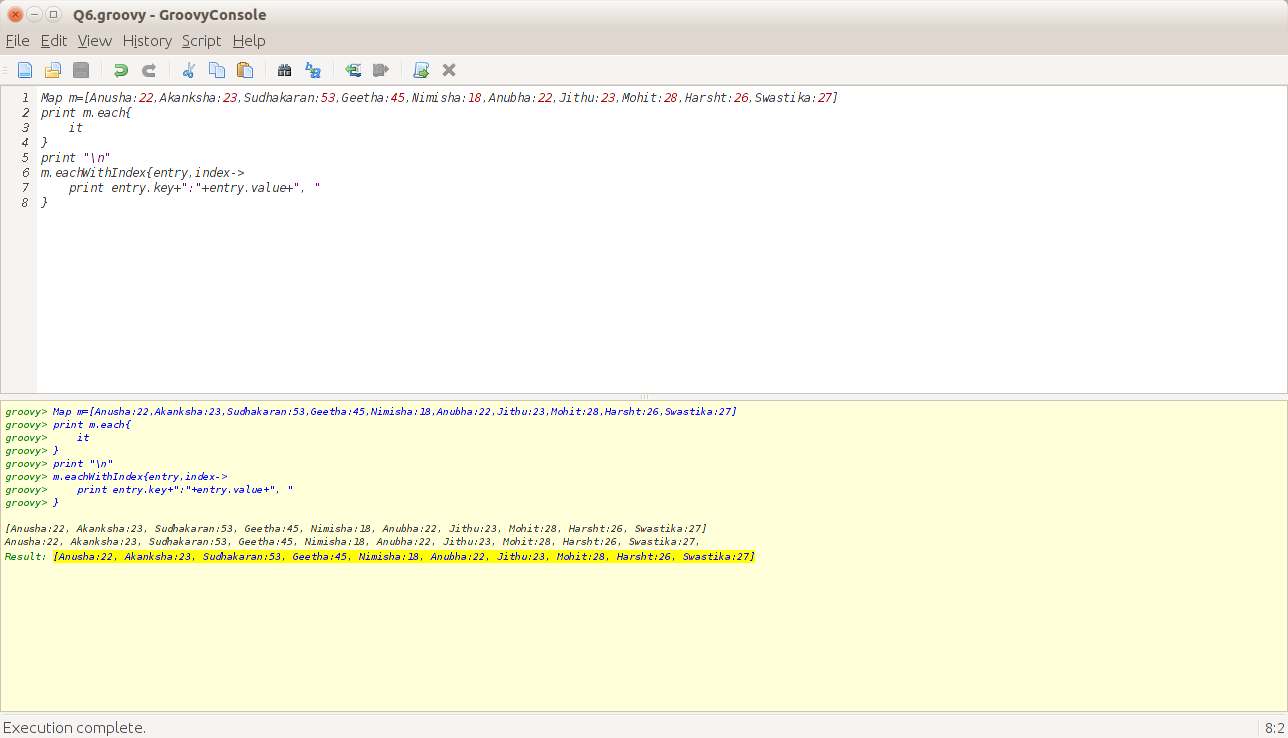
}

print "\n"

m.eachWithIndex{entry,index->

print entry.key+":"+entry.value+", "

}



* Create a new map by adding two existing maps

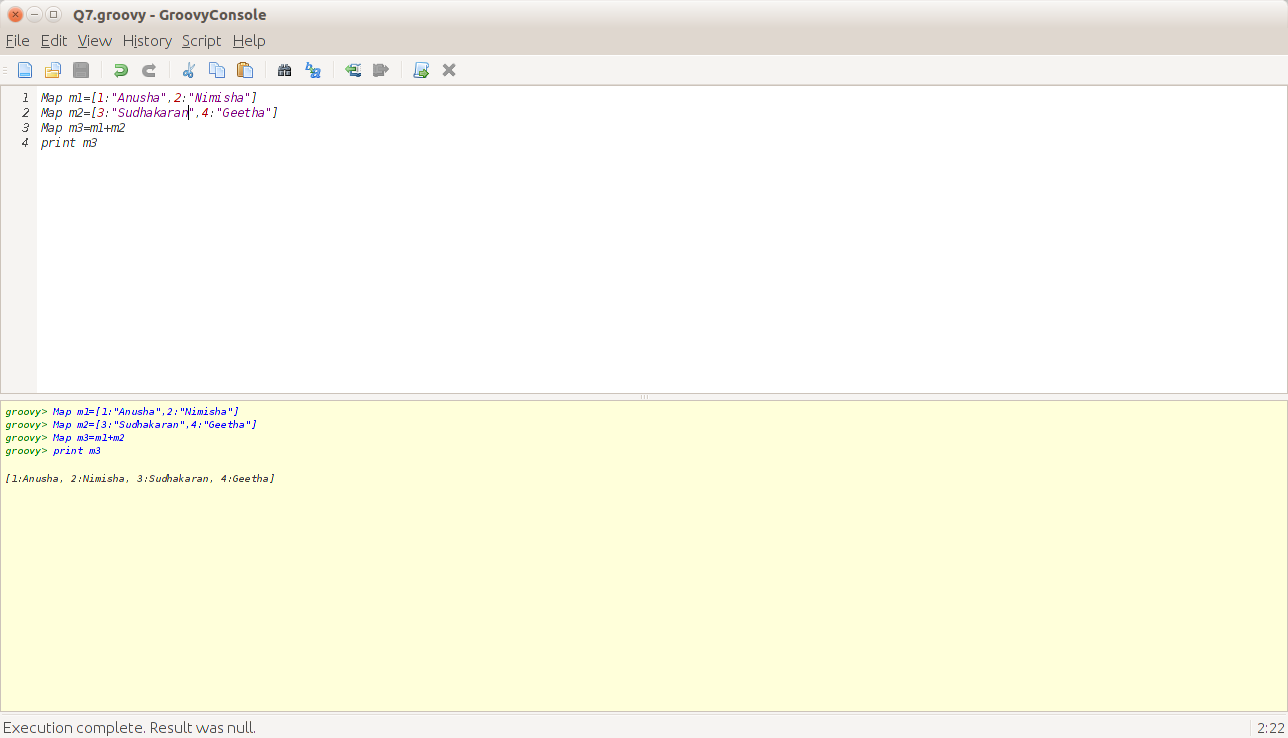
Sol:-

Map m1=[1:"Anusha",2:"Nimisha"]

Map m2=[3:"Sudhakaran",4:"Geetha"]

Map m3=m1+m2

print m3



* Try the following code on a map:

**println map.class**

**println map.getClass()**

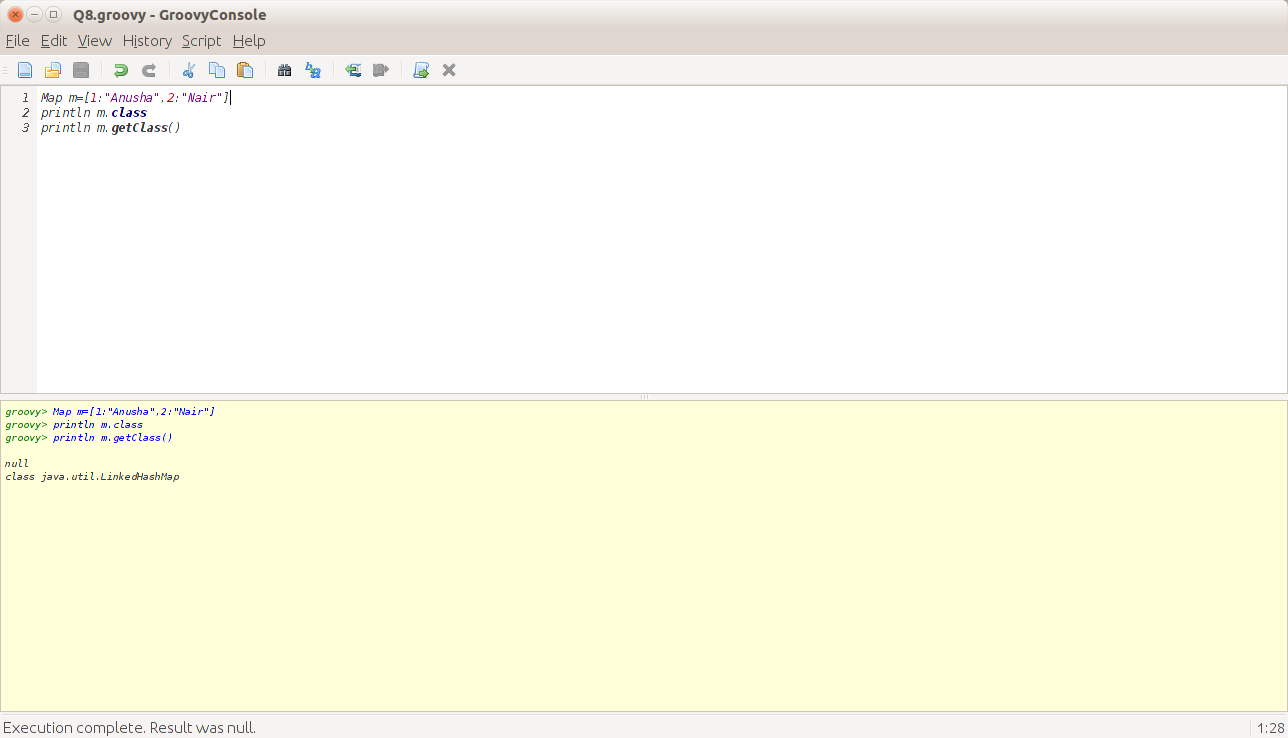
What do you observe?

Sol:-

Map m=[1:"Anusha",2:"Nair"]

println m.class

println m.getClass()



* Consider the following map:

**Map m = ['1' : 2, '2' : 3, '3' : 4, '2':5]**

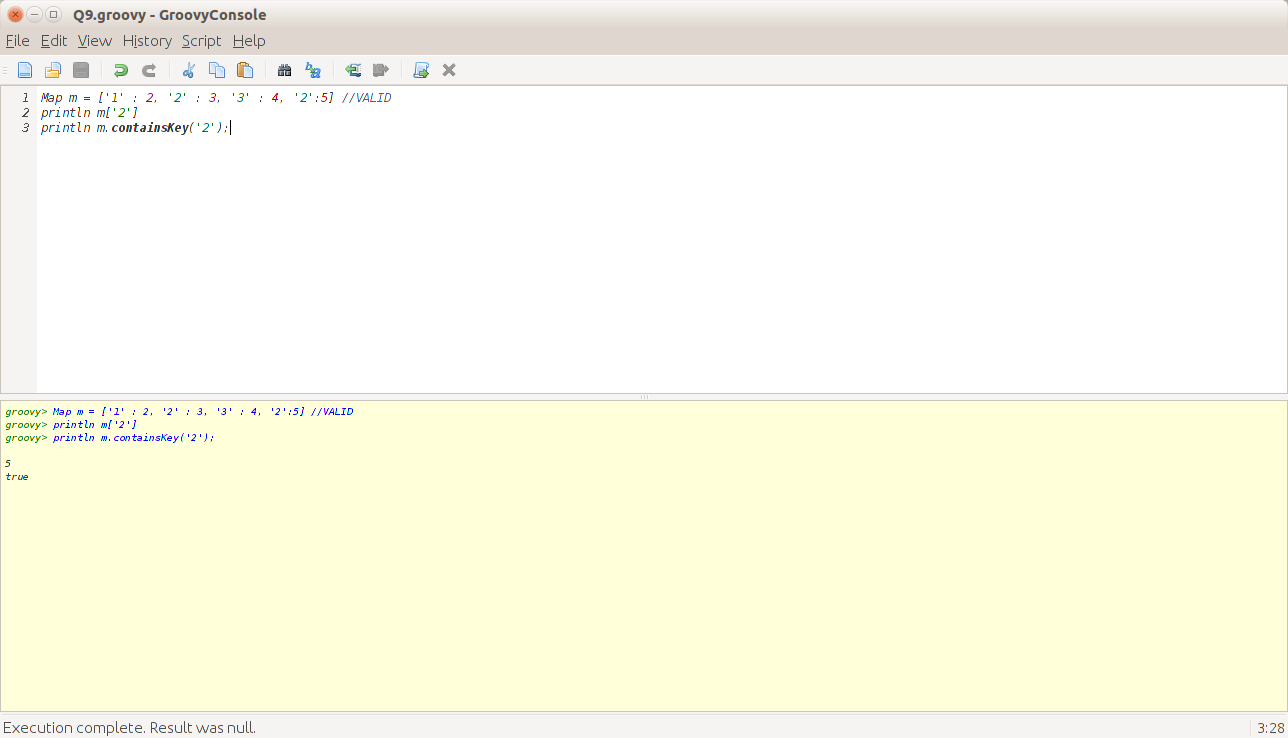
Is this a valid construction? What is the value of m['2']?

Sol:-

Map m = ['1' : 2, '2' : 3, '3' : 4, '2':5] //VALID

println m['2']

println m.containsKey('2');



* Consider the following map:

**Map m = [**

     ‘**Computing’ : [‘Computing’ : 600, ‘Information Systems’ : 300],**

         ‘**Engineering’ : [‘Civil’ : 200, ‘Mechanical’ : 100],**

         ‘**Management’ : [‘Management’ : 800]**

**]**

a) How many university departments are there?

b) How many programs are delivered by the Computing department?

c) How many students are enrolled in the Civil Engineering program?

Sol:-

Map<String,Map> m=['Computing':['Computing':600,'InformationSystems':300],

'Engineering':['Civil':200,'Mechanical':100],

'Management':['Management':800]

]

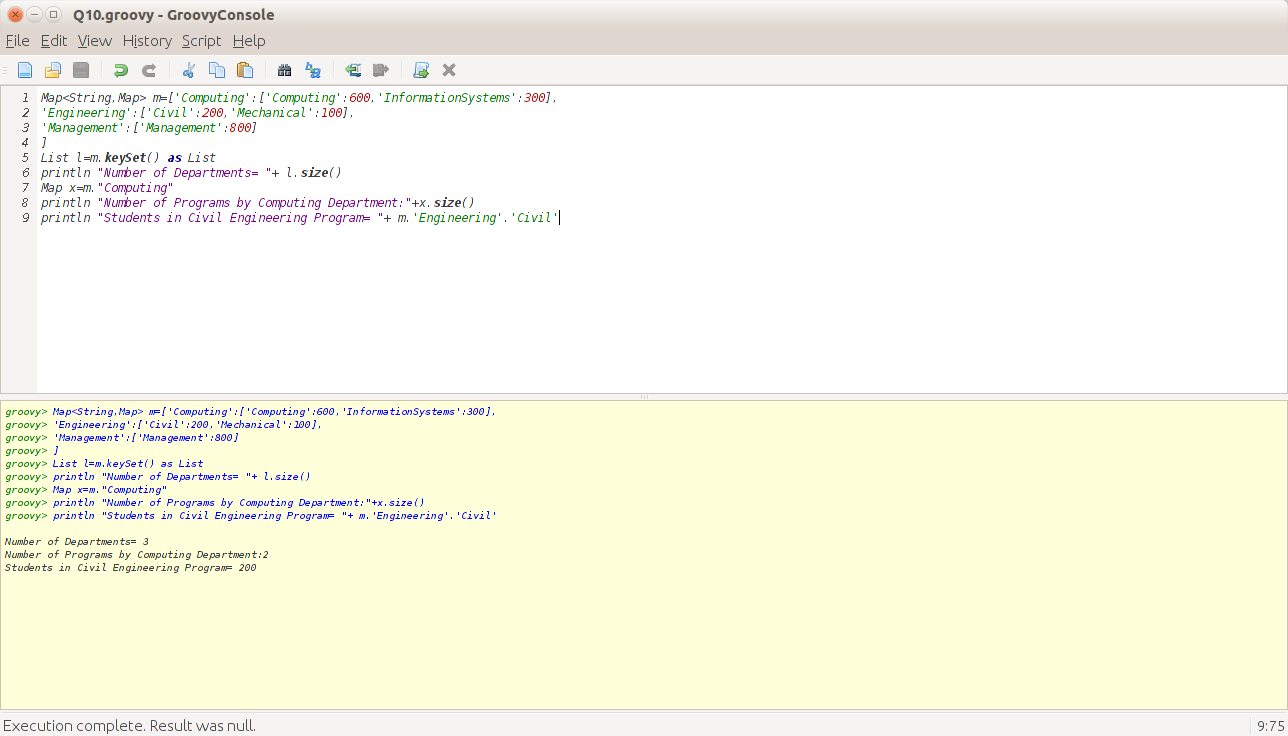
List l=m.keySet() as List

println "Number of Departments= "+ l.size()

Map x=m."Computing"

println "Number of Programs by Computing Department:"+x.size()

println "Students in Civil Engineering Program= "+ m.'Engineering'.'Civil'



Consider a class named "Employee" which has the following properties:

1) Name

2) Age

3) DepartmentName

4) EmployeeNumber

5) Salary

Let's say that there's a list of 50 employees available. Perform the following operations on the list of employees:

a) Group the employees on the basis of the bracket in which their salary falls. The ranges are 0-5000, 5001 and 10000, and so on.

b) Get a count of the number of employees in each department

c) Get the list of employees whose age is between 18 and 35

d) Group the employees according to the alphabet with which their first name starts and display the number of employees in each group whose age is greater than 20

e) Group the employees according to their department.

Sol:-

class Employee

{

String Name;

byte age;

String dept;

int e\_no;

int salary;

}

Employee e1=new Employee(name:'Anusha',age:23 as byte,dept:'Finance',e\_no:101,salary:20000)

Employee e2=new Employee(name:'Nimisha',age:53 as byte,dept:'Marketing',e\_no:102,salary:2000)

Employee e3=new Employee(name:'Geetha',age:83 as byte,dept:'Manufacturing',e\_no:103,salary:3000)

Employee e4=new Employee(name:'Sudhakaran',age:13 as byte,dept:'Marketing',e\_no:104,salary:4000)

Employee e5=new Employee(name:'Mohit',age:53 as byte,dept:'Manufacturing',e\_no:105,salary:6000)

List l=[e1,e2,e3,e4,e5]

println "Sol1"

Map m1=l.groupBy({

if((it.salary>=0) && (it.salary<=5000))

return "Within 5000";

else if((it.salary>5000) && (it.salary<=10000))

return "Within 10000";

else

return "above 10000"

})

println m1;

println "Sol2"

Map m2=l.groupBy({it.dept})

m2.each({

print "\t"+it.key+" "+it.value.size();

})

print "\n"

println "Sol3"

Map m3=l.groupBy({

if(it.age>18 && it.age<35)

return "Adults"

else

return "Aged"})

println m3

println "Sol4"

Map m=l.groupBy{it.name[0]};

println m

m.each{

print "Name Start with:${it.key} and number of Employees with Age >20 are:"

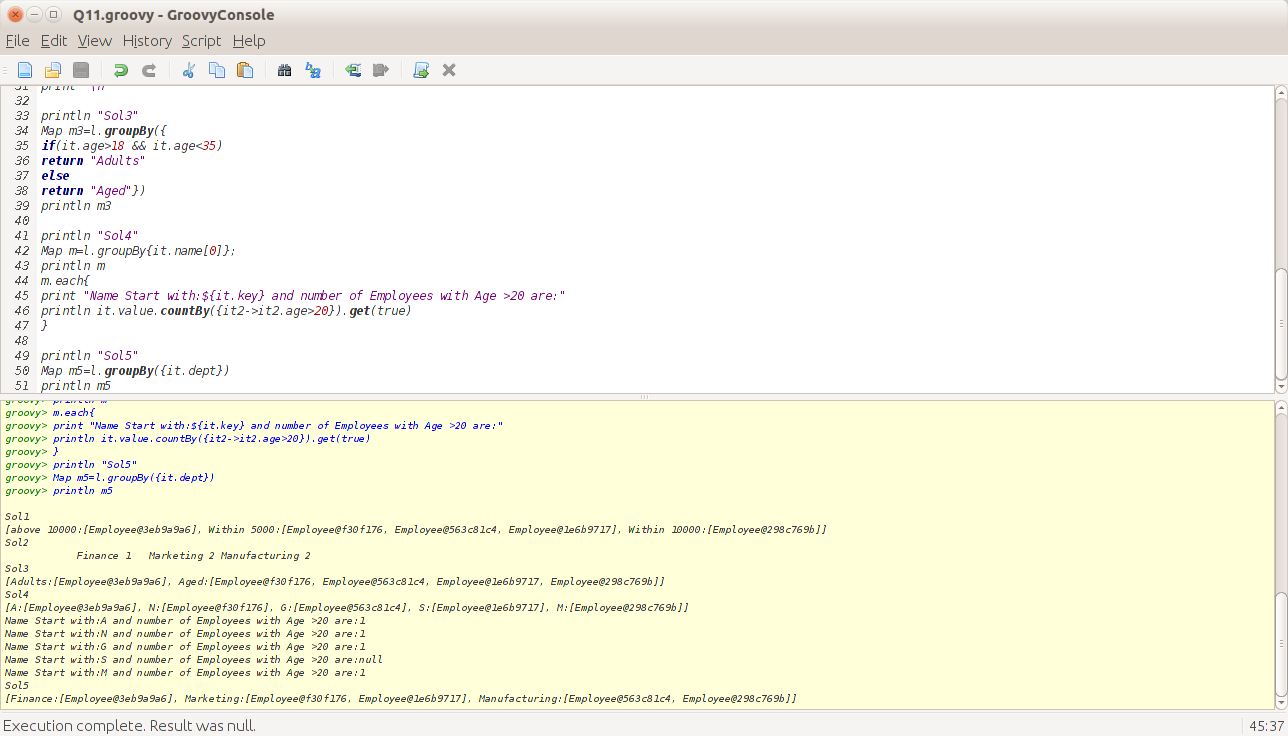
println it.value.countBy({it2->it2.age>20}).get(true)

}

println "Sol5"

Map m5=l.groupBy({it.dept})

println m5



Write a method which returns the value of passed key from a search string of the form  "http://www.google.com?name="

Sol:-

String str= "http://www.google.com?name=anusha&age=23&hobby=reading"

def var=str.tokenize("?")

def key

var.each{

key=it.split("&")

}

key.each{

println it

}

