

Assignment 16.1

Task 1

Create a calculator to work with rational numbers.

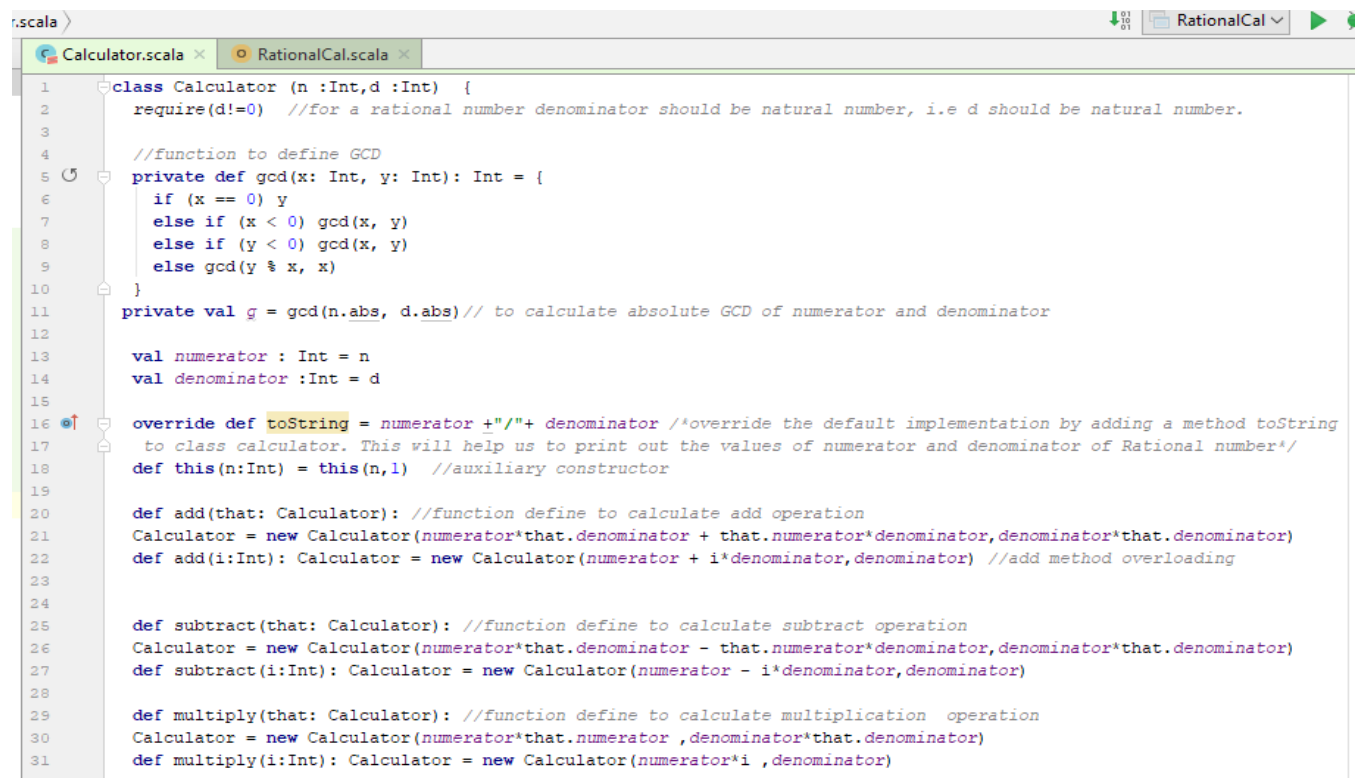
Requirements:

- It should provide capability to add, subtract, divide and multiply rational Numbers.
- Create a method to compute GCD (this will come in handy during operations on Rational).

Add option to work with whole numbers, which are also rational numbers i.e. (n/1)

- Achieve the above using auxiliary constructors
- Enable method overloading to enable each function to work with numbers and rational.

Screen shot show the program for calculator to work with rational number:



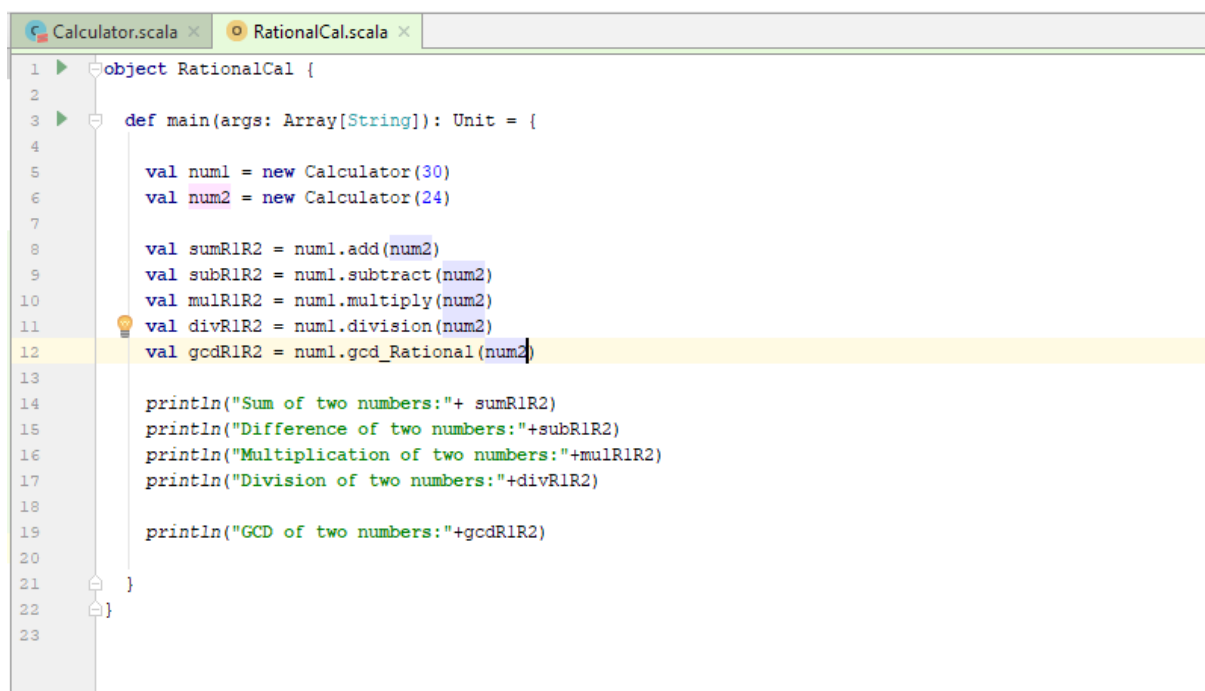
```
1 class Calculator (n :Int,d :Int) {
2   require(d!=0) //for a rational number denominator should be natural number, i.e d should be natural number.
3
4   //function to define GCD
5   private def gcd(x: Int, y: Int): Int = {
6     if (x == 0) y
7     else if (x < 0) gcd(x, y)
8     else if (y < 0) gcd(x, y)
9     else gcd(y % x, x)
10  }
11  private val g = gcd(n.abs, d.abs) // to calculate absolute GCD of numerator and denominator
12
13  val numerator : Int = n
14  val denominator :Int = d
15
16  override def toString = numerator + "/" + denominator /*override the default implementation by adding a method toString
17   to class calculator. This will help us to print out the values of numerator and denominator of Rational number*/
18  def this(n:Int) = this(n,1) //auxiliary constructor
19
20  def add(that: Calculator): //function define to calculate add operation
21  Calculator = new Calculator(numerator*that.denominator + that.numerator*denominator,denominator*that.denominator)
22  def add(i:Int): Calculator = new Calculator(numerator + i*denominator,denominator) //add method overloading
23
24
25  def subtract(that: Calculator): //function define to calculate subtract operation
26  Calculator = new Calculator(numerator*that.denominator - that.numerator*denominator,denominator*that.denominator)
27  def subtract(i:Int): Calculator = new Calculator(numerator - i*denominator,denominator)
28
29  def multiply(that: Calculator): //function define to calculate multiplication operation
30  Calculator = new Calculator(numerator*that.numerator ,denominator*that.denominator)
31  def multiply(i:Int): Calculator = new Calculator(numerator*i ,denominator)
32 }
```

```

32
33
34     def division(that: Calculator): //function define to calculate division operation
35     Calculator = new Calculator(numerator*that.denominator , denominator*that.numerator)
36     def division(i:Int): Calculator = new Calculator(numerator , denominator*i)
37
38
39     def gcd_Rational(that: Calculator): //function define to calculate GCD operation for two rational numbers
40     Calculator = new Calculator((gcd(numerator*that.denominator, denominator*that.numerator))/(denominator*that.denominator))
41     def gcd_rational(i:Int) : Calculator = new Calculator(gcd(numerator/denominator, i))
42 }
43

```

Screen shot below shows the object created for add, subtract, multiplication, division and gcd on rational numbers.

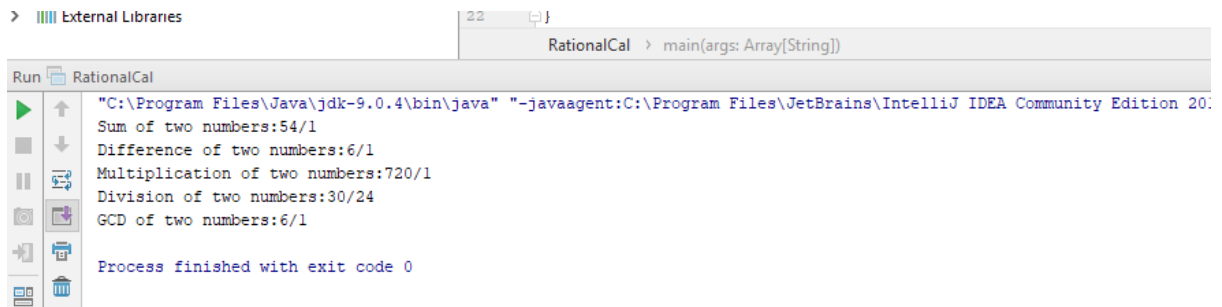


```

Calculator.scala x RationalCal.scala x
1  object RationalCal {
2
3  def main(args: Array[String]): Unit = {
4
5      val num1 = new Calculator(30)
6      val num2 = new Calculator(24)
7
8      val sumR1R2 = num1.add(num2)
9      val subR1R2 = num1.subtract(num2)
10     val mulR1R2 = num1.multiply(num2)
11     val divR1R2 = num1.division(num2)
12     val gcdR1R2 = num1.gcd_Rational(num2)
13
14     println("Sum of two numbers:" + sumR1R2)
15     println("Difference of two numbers:" + subR1R2)
16     println("Multiplication of two numbers:" + mulR1R2)
17     println("Division of two numbers:" + divR1R2)
18
19     println("GCD of two numbers:" + gcdR1R2)
20
21 }
22
23

```

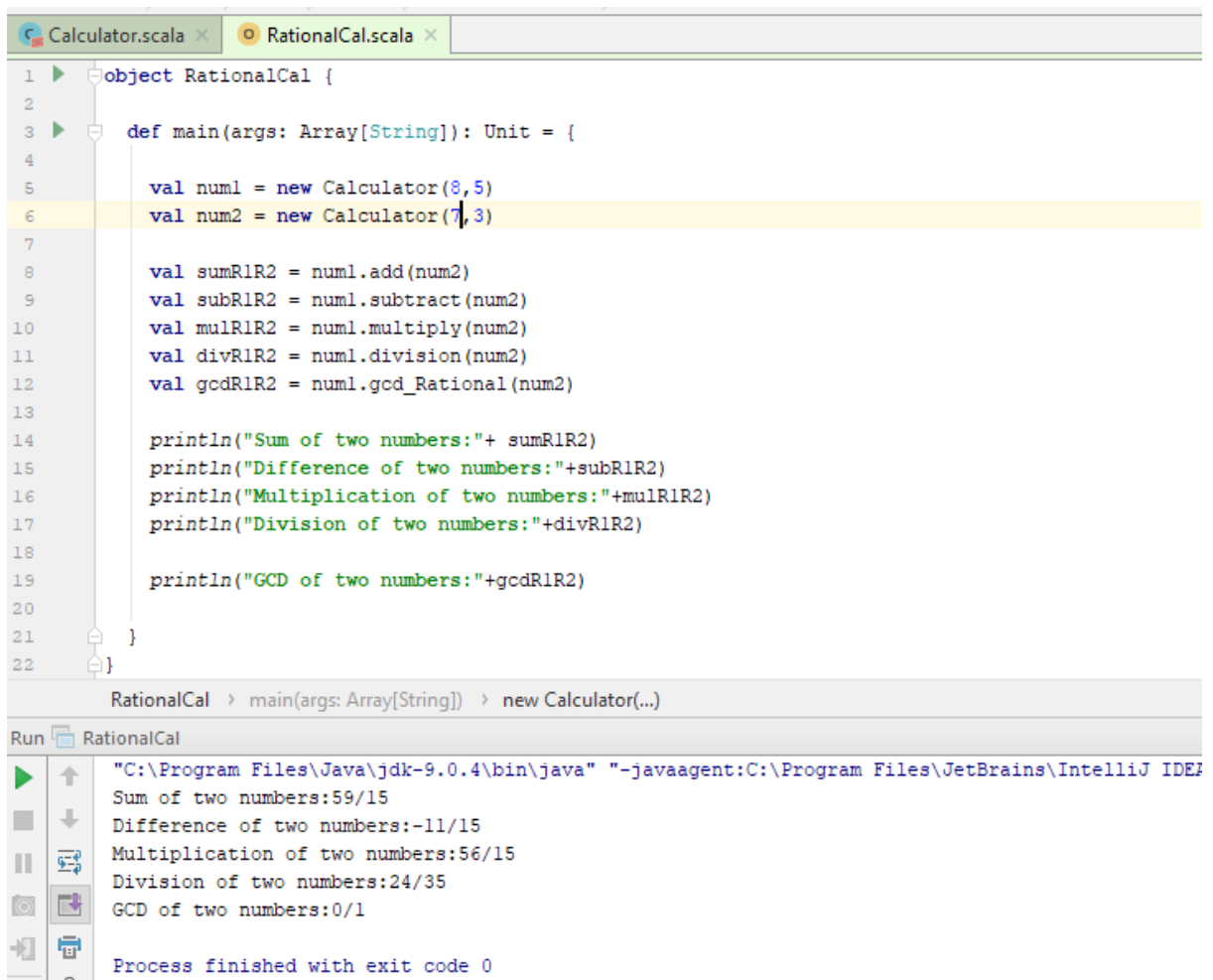
Sample output of above object:



The screenshot shows the Run console in IntelliJ IDEA. The top bar indicates the current context is 'RationalCal > main(args: Array[String])'. The console output for 'Run RationalCal' is as follows:

```
"C:\Program Files\Java\jdk-9.0.4\bin\java" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2017.1\lib\idea_rt.jar=60281:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2017.1\bin" -Dfile.encoding=UTF-8
Sum of two numbers:54/1
Difference of two numbers:6/1
Multiplication of two numbers:720/1
Division of two numbers:30/24
GCD of two numbers:6/1
Process finished with exit code 0
```

Output with rational numbers, num1=8/5 and num2=7/3



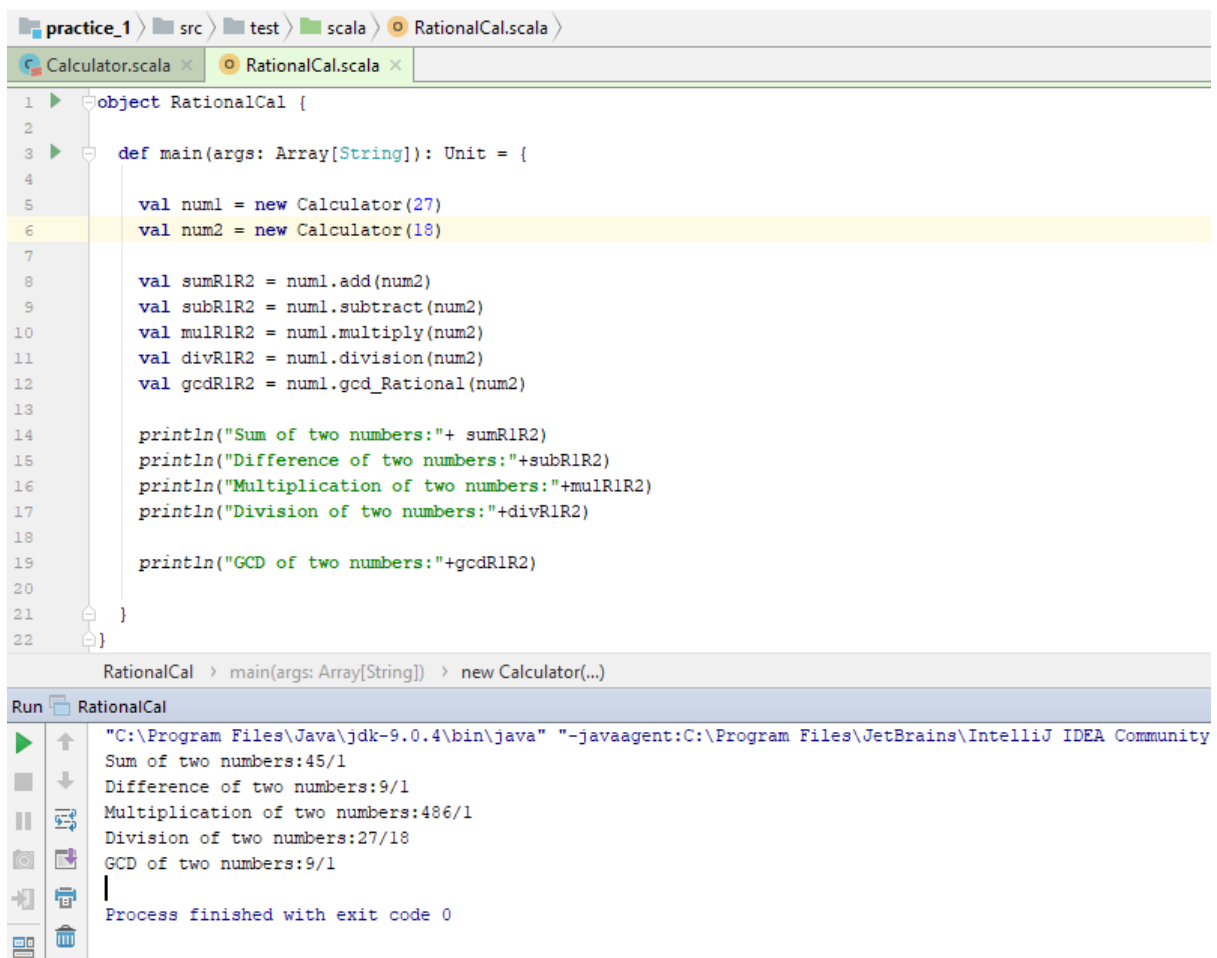
The screenshot shows the source code of 'RationalCal.scala' in the editor. The code defines an object 'RationalCal' with a 'main' method that creates two 'Calculator' objects and performs various operations on them. The Run console below shows the output of the program.

```
1 object RationalCal {
2
3 def main(args: Array[String]): Unit = {
4
5     val num1 = new Calculator(8,5)
6     val num2 = new Calculator(7,3)
7
8     val sumR1R2 = num1.add(num2)
9     val subR1R2 = num1.subtract(num2)
10    val mulR1R2 = num1.multiply(num2)
11    val divR1R2 = num1.division(num2)
12    val gcdR1R2 = num1.gcd_Rational(num2)
13
14    println("Sum of two numbers:"+ sumR1R2)
15    println("Difference of two numbers:"+subR1R2)
16    println("Multiplication of two numbers:"+mulR1R2)
17    println("Division of two numbers:"+divR1R2)
18
19    println("GCD of two numbers:"+gcdR1R2)
20
21 }
22 }
```

The Run console output for 'RationalCal' is as follows:

```
"C:\Program Files\Java\jdk-9.0.4\bin\java" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2017.1\lib\idea_rt.jar=60281:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2017.1\bin" -Dfile.encoding=UTF-8
Sum of two numbers:59/15
Difference of two numbers:-11/15
Multiplication of two numbers:56/15
Division of two numbers:24/35
GCD of two numbers:0/1
Process finished with exit code 0
```

Output with rational numbers, num1=27 and num2= 18



The screenshot displays the IntelliJ IDEA IDE with a Scala project named 'practice_1'. The 'src' directory contains a file 'RationalCal.scala'. The code defines an object 'RationalCal' with a 'main' method that takes an array of strings as input. Inside the 'main' method, two 'Calculator' objects are created: 'num1' with value 27 and 'num2' with value 18. The code then performs several operations: addition, subtraction, multiplication, division, and GCD calculation, with the results printed to the console. The output window shows the following results: 'Sum of two numbers:45/1', 'Difference of two numbers:9/1', 'Multiplication of two numbers:486/1', 'Division of two numbers:27/18', and 'GCD of two numbers:9/1'. The process finished with exit code 0.

```
1 object RationalCal {
2
3   def main(args: Array[String]): Unit = {
4
5     val num1 = new Calculator(27)
6     val num2 = new Calculator(18)
7
8     val sumR1R2 = num1.add(num2)
9     val subR1R2 = num1.subtract(num2)
10    val mulR1R2 = num1.multiply(num2)
11    val divR1R2 = num1.division(num2)
12    val gcdR1R2 = num1.gcd_Rational(num2)
13
14    println("Sum of two numbers:"+ sumR1R2)
15    println("Difference of two numbers:"+subR1R2)
16    println("Multiplication of two numbers:"+mulR1R2)
17    println("Division of two numbers:"+divR1R2)
18
19    println("GCD of two numbers:"+gcdR1R2)
20
21  }
22 }
```

Run RationalCal

"C:\Program Files\Java\jdk-9.0.4\bin\java" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community

Sum of two numbers:45/1
Difference of two numbers:9/1
Multiplication of two numbers:486/1
Division of two numbers:27/18
GCD of two numbers:9/1
|
Process finished with exit code 0