Session 14

Assignment 1

|  |  |
| --- | --- |
| **Prepared For:** | AcadGild |
|  |  |
| **Document Approval:** | **AcadGild** |
|  |  |
|  |  |
|  |  |
|  |  |
| **Project Title:** | Session 14 – Assignment 1 |
|  |  |
| **Prepared By:** | Duncan Burgess |
|  |  |
|  | dburgess@duncb.com |
|  |  |
| **Primary Engineer:** | Duncan Burgess |
|  |  |
| **Document Reference:** | **Session 14 – Assignment 1** |
|  |  |
| **Start Date:** | 10/10/2017 |
|  |  |
|  |  |



# 

# Contents

[Contents 2](#_Toc495674311)

[Change History 3](#_Toc495674312)

[1. Problem Statement 4](#_Toc495674313)

[2. Solution 4](#_Toc495674314)

[2.1. Rational class is defined 4](#_Toc495674315)

[2.2. object class "Calculator" is defined 4](#_Toc495674316)

[2.3. Code created 4](#_Toc495674317)

[2.4. Running in Eclipse 7](#_Toc495674318)

[2.5. The results 9](#_Toc495674319)

# Change History

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Document Revision** | **Date** | **Authored By** | **Authorised By** | **Sections Affected** | **Reason for Change** |
| Rev 01 | 13/10/2017 | Duncan Burgess |  | All | Initial release. |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# Problem Statement

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.

# Solution

## Rational class is defined

* one primary constructor with two parameters is defined (defines class)
* one auxiliary constructor is defined
* two data members are defined "numer" and "denom" (for numerator and denominator)
* Check value entered denominator is 0, if it is zero then exception is generated with message
* GCD method is defined which outputs smallest fraction
* method overloading is used to overload functions to work with both rational and integer numbers

## object class "Calculator" is defined

* Number Choice() method is defined which gives two options to input i.e. whether user wants to work with rational number or with integer (whole) number
* [NOTE -> user can enter + or - values both for rational and whole numbers]
* make Rational() method is defined which creates a rational number if rational number is chosen.
* Options() method is defined which provides options of operations to user i.e. add, subtract, multiply, divide
* Compute() method is defined where input(s) from user is taken either for rational or for integer (whole) number call to overloaded =function depending upon chosen
* inside main() method, do while loop is run until user presses 'y' to exit, Or else the user can continue to perform operation retaining the value of the previous operation

## Code created

***package*** *com.duncb.spark{*

***import*** *scala.io.StdIn.\_ //Required for input*

***class*** *Rational(n: Int, d: Int) {*

*require(d!=0,"den1inator cannot be 0")*

***private******val*** *g = gcd(n.abs, d.abs)*

***val*** *num1 = n / g*

***val*** *den1 = d / g*

*//This is the auxiliary constructor*

***def******this****(n: Int) =* ***this****(n, 1)*

*//gcd method*

***private******def*** *gcd(a: Int, b: Int): Int =*

***if*** *(b == 0) a* ***else*** *gcd(b, a % b)*

*//overloaded methods*

***def*** *+ (that: Rational): Rational =*

***new*** *Rational(*

*num1 \* that.den1 + that.num1 \* den1,*

*den1 \* that.den1*

*)*

***def*** *+ (i: Int): Rational =*

***new*** *Rational(num1 + i \* den1, den1)*

***def*** *- (that: Rational): Rational =*

***new*** *Rational(*

*num1 \* that.den1 - that.num1 \* den1,*

*den1 \* that.den1*

*)*

***def*** *- (i: Int): Rational =*

***new*** *Rational(num1 - i \* den1, den1)*

***def*** *\* (that: Rational): Rational =*

***new*** *Rational(num1 \* that.num1, den1 \* that.den1)*

***def*** *\* (i: Int): Rational =*

***new*** *Rational(num1 \* i, den1)*

***def*** */ (that: Rational): Rational =*

***new*** *Rational(num1 \* that.den1, den1 \* that.num1)*

***def*** */ (i: Int): Rational =*

***new*** *Rational(num1, den1 \* i)*

*}*

***object*** *Calculator {*

*//To choose a rational or a whole number*

***def*** *NumberChoice() = {*

*println("Please enter your choice.of Number.")*

*println("1. Rational")*

*println("2. Whole")*

*}*

*//creates a rational number*

***def*** *makeRational(rational:Rational):Rational = {*

*println("Enter num1ator and den1inator : ")*

***val*** *p = readInt()*

***val*** *q = readInt()*

*rational.+(****new*** *Rational(p, q))*

*}*

*//provides option of operations to user*

***def*** *Options() = {*

*println("======== Dunkies Calculator ========")*

*println(" ")*

*println("1. Add a rational")*

*println("2. Subtract by rational")*

*println("3. Multiply by rational")*

*println("4. Divide byrational")*

*println("5. Add an integer")*

*println("6. Subtract by integer")*

*println("7. Multiply by integer")*

*println("8. Divide by integer")*

*println(" y to exit")*

*}*

*//Overload method*

***def*** *Compute(rational: Rational, input: Int): Rational = {*

*input* ***match*** *{*

***case*** *1 =>*

*println("Enter num1ator and den1inator of a rational number to add : ")*

***val*** *p = readInt()*

***val*** *q = readInt()*

*rational.+(****new*** *Rational(p, q))*

***case*** *2 =>*

*println("Enter num1ator and den1inator of a rational number to subtract : ")*

***val*** *p = readInt()*

***val*** *q = readInt()*

*rational.-(****new*** *Rational(p, q))*

***case*** *3 =>*

*//val p = scala.io.StdIn.readInt()*

*//val q = scala.io.StdIn.readInt()*

*println("Enter num1ator and den1inator of a rational number to multiply")*

***val*** *p = readInt()*

***val*** *q = readInt()*

*rational.\*(****new*** *Rational(p, q))*

***case*** *4 =>*

*//val p = scala.io.StdIn.readInt()*

*println("Enter num1ator and den1inator of a rational number to divide")*

***val*** *p = readInt()*

***val*** *q = readInt()*

*rational./(****new*** *Rational(p,q))*

***case*** *5 =>*

*//val p = scala.io.StdIn.readInt()*

*println("Enter an integer to add ")*

***val*** *p = readInt()*

*rational.+(****new*** *Rational(p))*

***case*** *6 =>*

*//val p = scala.io.StdIn.readInt()*

*println("Enter an integer to subtract : ")*

***val*** *p = readInt()*

*rational.-(****new*** *Rational(p))*

***case*** *7 =>*

*//val p = scala.io.StdIn.readInt()*

*println("Enter an integer to multiply: ")*

***val*** *p = readInt()*

*rational.\*(****new*** *Rational(p))*

***case*** *8 =>*

*//val p = scala.io.StdIn.readInt()*

*println("Enter a non-zero integer to divide ")*

***val*** *p = readInt()*

*rational./(****new*** *Rational(p))*

***case*** *\_ =>*

*rational*

*}*

*}*

*//runs loop until user types 'y' to finish*

***def*** *main(args: Array[String]): Unit = {*

*// def public = static main main(String[] args)*

***var*** *rnum1:Rational =* ***new*** *Rational(0)*

***var*** *rnum2:Rational =* ***new*** *Rational(0)*

***var*** *num = 0*

***var*** *choice1 = 0*

***var*** *choice2 = 0*

***var*** *ch = 'y'*

*println("======== Dunkies Calculator =======")*

*NumberChoice()*

*choice1 = readInt()*

*choice1* ***match*** *{ //matching choice*

***case*** *1 =>*

*rnum1 = makeRational(rnum1)*

*println("Rational Number is : "+rnum1.toString)*

*rnum2 = rnum1*

***do*** *{*

*Options()*

*println("Please enter any of the above options : ")*

*choice2 = readInt()*

*//rnum2 = Compute(rnum2, choice2)*

*//rnum2 = rnum1*

*rnum2 = Compute(rnum2, choice2)*

***if****(rnum2.den1 == 1)*

*println("Output is : " +rnum2.num1)*

***else******if****(rnum2.den1 < 0)*

*println("Output is : " + "-"+ rnum2.num1+"/"+rnum2.den1.abs)*

***else***

*//println("Output is : " + rnum2.toString)*

*println("Output is : " + rnum2.num1+"/"+rnum2.den1)*

*println("Press y to finish or press other key to continue...")*

*ch = readChar()*

*}****while*** *(ch != 'y')*

***case*** *2 =>*

*println("Enter a number : ")*

*num = readInt()*

*rnum2 =* ***new*** *Rational(num)*

***do*** *{*

*Options()*

*println("Please enter any of the above options : ")*

*choice2 = readInt()*

*rnum2 = Compute(rnum2, choice2)*

***if****(rnum2.den1 == 1)*

*println("Output is : "+rnum2.num1)*

***else******if****(rnum2.den1 < 0)*

*println("Output is : " + "-"+ rnum2.num1+"/"+rnum2.den1.abs)*

***else***

*//println("Output is : " + rnum2.toString)*

*println("Output is : " + rnum2.num1+"/"+rnum2.den1)*

*println("Press y to finish or press other key to continue...")*

*ch = readChar()*

*}****while*** *(ch != 'y')*

*}*

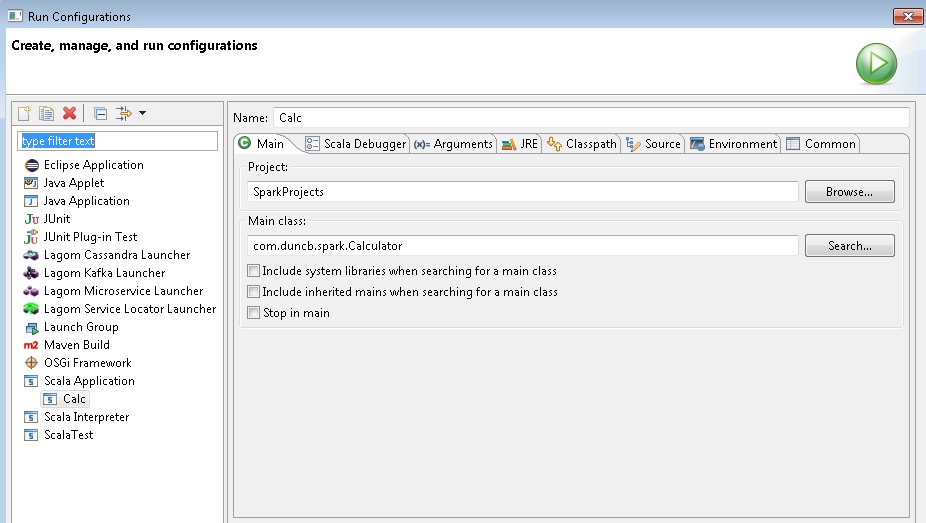
*}*

*}*

*}*

## Running in Eclipse

After importing the required jars



## The results

*======== Dunkies Calculator =======*

*Please enter your choice.of Number.*

*1. Rational*

*2. Whole*

*1*

*Enter num1ator and den1inator :*

*9*

*2*

*Rational Number is : 4.5/1*

*======== Dunkies Calculator ========*

*1. Add a rational*

*2. Subtract by rational*

*3. Multiply by rational*

*4. Divide byrational*

*5. Add an integer*

*6. Subtract by integer*

*7. Multiply by integer*

*8. Divide by integer*

*y to exit*

*Please enter any of the above options :*

*2*

*Enter num1ator and den1inator of a rational number to subtract :*

*8*

*9*

*Output is : 65/18*

*Press y to finish or press other key to continue...*

*g*

*======== Dunkies Calculator ========*

*1. Add a rational*

*2. Subtract by rational*

*3. Multiply by rational*

*4. Divide byrational*

*5. Add an integer*

*6. Subtract by integer*

*7. Multiply by integer*

*8. Divide by integer*

*y to exit*

*Please enter any of the above options :*

*3*

*Enter num1ator and den1inator of a rational number to multiply*

*6*

*4*

*Output is : 65/12*

*Press y to finish or press other key to continue...*

*u*

*======== Dunkies Calculator ========*

*1. Add a rational*

*2. Subtract by rational*

*3. Multiply by rational*

*4. Divide byrational*

*5. Add an integer*

*6. Subtract by integer*

*7. Multiply by integer*

*8. Divide by integer*

*y to exit*

*Please enter any of the above options :*

*4*

*Enter num1ator and den1inator of a rational number to divide*

*8*

*13*

*Output is : 845/96*

*Press y to finish or press other key to continue...*

*u*

*======== Dunkies Calculator ========*

*1. Add a rational*

*2. Subtract by rational*

*3. Multiply by rational*

*4. Divide byrational*

*5. Add an integer*

*6. Subtract by integer*

*7. Multiply by integer*

*8. Divide by integer*

*y to exit*

*Please enter any of the above options :*

*5*

*Enter an integer to add*

*54*

*Output is : 6029/96*

*Press y to finish or press other key to continue...*

*u*

*======== Dunkies Calculator ========*

*1. Add a rational*

*2. Subtract by rational*

*3. Multiply by rational*

*4. Divide byrational*

*5. Add an integer*

*6. Subtract by integer*

*7. Multiply by integer*

*8. Divide by integer*

*y to exit*

*Please enter any of the above options :*

*6*

*Enter an integer to subtract :*

*32*

*Output is : 2957/96*

*Press y to finish or press other key to continue...*

*u*

*======== Dunkies Calculator ========*

*1. Add a rational*

*2. Subtract by rational*

*3. Multiply by rational*

*4. Divide byrational*

*5. Add an integer*

*6. Subtract by integer*

*7. Multiply by integer*

*8. Divide by integer*

*y to exit*

*Please enter any of the above options :*

*7*

*Enter an integer to multiply:*

*5*

*Output is : 14785/96*

*Press y to finish or press other key to continue...*

*u*

*======== Dunkies Calculator ========*

*1. Add a rational*

*2. Subtract by rational*

*3. Multiply by rational*

*4. Divide byrational*

*5. Add an integer*

*6. Subtract by integer*

*7. Multiply by integer*

*8. Divide by integer*

*y to exit*

*Please enter any of the above options :*

*8*

*Enter a non-zero integer to divide*

*15*

*Output is : 2957/288*

*Press y to finish or press other key to continue...*

*u*

*======== Dunkies Calculator ========*

*1. Add a rational*

*2. Subtract by rational*

*3. Multiply by rational*

*4. Divide byrational*

*5. Add an integer*

*6. Subtract by integer*

*7. Multiply by integer*

*8. Divide by integer*

*y to exit*

*Please enter any of the above options :*

*3*

*Enter num1ator and den1inator of a rational number to multiply*

*1*

*2*

*Output is : 2957/576*

*Press y to finish or press other key to continue...*

*y*