Lab #3

Notes

Folder organization for solutions on the student's account:

```
z:\
|
|--Programming02
|----Lab01Problem01
|----Lab01Problem02
|----…
|----Lab02Problem01
|----…
|--Programming01
```

Task #1: "Calculator of Rational Numbers" (2.5%)

Create application which can help users to do arithmetic operations (addition, subtraction, multiplication, division, comparisons) with rational numbers. User enters first rational, operator (+, -, *, /, <, >, =, !=, <=, >=), second rational and, after that, application prints the result:

```
Calculator of Rational numbers
Enter 1st rational (Ctrl-Z: exit): 1/2
Enter operator: +
Enter 2nd rational (Ctrl-Z: exit): 2/3
Result: 1/2 + 2/3 = 7/6
Enter 1st rational (Ctrl-Z: exit): one
Incorrect rational: one
Enter 1st rational(Ctrl-Z: exit): 1/0
Incorrect rational: Denominator can't be zero
Enter 1st rational: ^Z
```

Application has to check all data it gets from user carefully and in the case of incorrect user input it has to print appropriate message. User can stop application entering Ctrl-Z (or Ctrl-D in Linux) instead of input data.

In this program you have to develop class Rational to represent rational numbers. Objects of this class (variables of this type) must support methods:

Method	Meaning
Rational(int num, int den)	Creates rational. Throws Exception if den is equal to 0.
Rational add(Rational other)	Returns the result of addition
Rational sub(Rational other)	Returns the result of subtraction
Rational mul(Rational other)	Returns the result of multiplication
Rational div(Rational other)	Returns the result of division
int compareTo(Rational other)	a.compareTo(b) returns -1 if a < b; 0 if a = b 1 if a > b
String toString()	Returns string representation of rational in following format: numerator/denominator
static Rational parse(String s)	This static method takes string representation of rational number and returns the corresponding number. Method has to throw Exception if parameter s is not a correct string representation of rational number.

Objects of class Rational must be always in reduced form. Keep the sign of the number in its numerator. Develop inner (private) methods to fix sign of rational , to reduce rational and to find greatest common divisor of numerator and denominator.

Notes

- **Try to check all possible errors in user's input**

Home Reading: Liang Introduction to Java Programming 8th ed. Chapter 10.