

+ getters, setters

+ toString, display

+ конструкторы

class NumFraction

package com.company;  
  
import java.util.Scanner;  
  
public class NumFraction *{* // поля класса NumFraction  
 private int numerator; // числитель  
 private int denominator; // знаменатель  
 private int full;// целая часть дроби  
  
 // getters  
 public int getNumerator*() {* return numerator;  
 *}* public int getDenominator*() {* return denominator;  
 *}* // setters  
 public void setNumerator*(*int value*) {* numerator = value;  
 *}* public void setDenominator*(*int value*) {* denominator = value;  
 *}* // Конструкторы  
 // 1. Конструктор без параметров - по умолчанию  
 public NumFraction*() {* // инициализация полей по умолчанию  
 numerator = 1;  
 denominator = 2;  
 *}* // 2. Конструктор с двумя параметрами int  
 public NumFraction*(*int \_numerator, int \_denominator*) {* numerator = \_numerator;  
 denominator = \_denominator;  
 *}* // Методы класса NumFraction  
 // Метод ввода дроби с клавиатуры через консоль  
 public void inputFraction*(*Scanner in*) {* while *(*true*) {* System.*out*.println*(*"Введите дробь:"*)*;  
 System.*out*.print*(*"Числитель = "*)*;  
 numerator = in.nextInt*()*;  
 System.*out*.print*(*"Знаменатель = "*)*;  
 denominator = in.nextInt*()*;  
 if *(*numerator == 0 || denominator < 1*)* System.*out*.println*(*"Введены неверные данные, введите число больше 0"*)*;  
 else break;  
 *}  
 }* // Метод вывода класса на консоль  
 public void displayFraction*() {* // сводим вывод объекта на консоль к выводу его строкового представления  
 System.*out*.print*(*toString*())*;  
 *}* // Метод представления класса в виде строки - перегрузка метода toString  
 @Override  
 public String toString*() {* return numerator + "/" + denominator;  
 *}* // Метод сложения дробей:  
 public NumFraction sum*(*NumFraction sum*) {* NumFraction res;  
 if *(*denominator == sum.denominator*) {* res = new NumFraction*(*numerator + sum.numerator, denominator*)*;  
 *}* else *{* res = new NumFraction*((*numerator \* sum.denominator*)* + *(*denominator \* sum.numerator*)* , denominator \* sum.denominator*)*;  
 *}* return res;  
 *}* // Метод вычитания дробей:  
 public NumFraction subtraction*(*NumFraction sub*) {* NumFraction res;  
 if *(*denominator == sub.denominator*) {* res = new NumFraction*(*numerator - sub.numerator, denominator*)*;  
  
 *}* else *{* res = new NumFraction*((*numerator \* sub.denominator*)* - *(*denominator \* sub.numerator*)* , denominator \* sub.denominator*)*;  
 *}* return res;  
 *}* // Метод умножения дробей:  
 public NumFraction multiplication*(*NumFraction mult*) {* NumFraction res;  
 res = new NumFraction*(*numerator \* mult.numerator, denominator \* mult.denominator*)*;  
  
 return res;  
 *}* // Метод деления дробей:  
 public NumFraction division*(*NumFraction div*) {* NumFraction res;  
 if *(*numerator > 0 && div.numerator > 0*) {* res = new NumFraction*(*numerator \* div.denominator, denominator \* div.numerator*)*;  
 *}* else *{* res = new NumFraction*(*numerator \* div.denominator \* *(*-1*)*, denominator \* Math.*abs(*div.numerator*))*;  
 *}* return res;  
 *}* // Метод выделения целой части дроби:  
 public void fullFraction*() {* if *(*numerator % denominator != 0*) {* full = numerator / denominator;  
 numerator %= denominator;  
 if *(*full != 0*) {* System.*out*.print*(*full + "(" + Math.*abs(*numerator*)* + "/" + denominator + ")"*)*;  
 *}* else if *(*numerator % denominator == 0*) {* full = numerator / denominator;  
 System.*out*.print*(*full*)*;  
 *}* else if *(*numerator > denominator*) {* full = numerator / denominator;  
 System.*out*.print*(*full*)*;  
 *}* else System.*out*.print*(*"дробь не имеет целую часть"*)*;  
 *}* else System.*out*.print*(*"дробь не имеет целую часть"*)*;  
 *}* //Метод перевода обыкновенной дроби в десятичную  
 public void decimal*() {* double res;  
 res = *(*double*)* numerator / *(*double*)* denominator + full;  
 System.*out*.print*(*" ≈ " + res*)*;  
 *}* //Метод нахождения НОД:  
 public int gcdNod*() {* int gcd = 1;  
 for *(*int i = 1; i <= Math.*abs(*numerator*)* && i <= denominator; i++*) {* if *(*numerator % i == 0 && denominator % i == 0*) {* gcd = i;  
 *}  
 }* return gcd;  
 *}* //Метод сокращения дроби:  
 public void simplify*() {* NumFraction res;  
 if *(*full != 0*) {* res = new NumFraction*(*Math.*abs(*numerator*)* / gcdNod*()*, denominator / gcdNod*())*;  
 System.*out*.print*(*full + "(" + res +")"*)*;  
 *}* else if *(*numerator == 0*) {* System.*out*.print*(*0*)*;  
 *}* else *{* res = new NumFraction*(*numerator / gcdNod*()*, denominator / gcdNod*())*;  
 System.*out*.print*(*res*)*;  
 *}  
 }  
 }*

class Main

package com.company;  
  
import java.util.Scanner;  
  
public class Main *{* public static void main*(*String*[]* args*) {* //1.Создание сканера, объявление переменных  
 Scanner in = new Scanner*(*System.*in)*;  
 NumFraction f = new NumFraction*()*;  
 NumFraction f1 = new NumFraction*()*;  
  
 //2.Вывод на экран  
 //Сложение дробей  
 f.inputFraction*(*in*)*;//ввод первой дроби  
 f1.inputFraction*(*in*)*;//ввод второй дроби  
 System.*out*.print*(*"\nСложение дробей: "*)*;//название арифметической операции  
 f.displayFraction*()*;//вывод первой дроби на экран  
 System.*out*.print*(*" + "*)*;  
 f1.displayFraction*()*;//вывод второй дроби на экран  
 System.*out*.print*(*" = "*)*;  
 //вызов метода сложения  
 NumFraction f2 = f.sum*(*f1*)*;  
 f2.displayFraction*()*;//вывод результата сложения на экран  
 System.*out*.print*(*"\nВыделение целой части: "*)*;  
 f2.fullFraction*()*;// выделение целой части дроби и вывод на экран  
 System.*out*.print*(*"\nСокращение дроби и перевод в десятичный вид: "*)*;  
 f2.simplify*()*;// сокращение дроби и вывод на экран  
 f2.decimal*()*;//перевод обыкновенной дроби в десятичную  
 System.*out*.println*()*;//перенос строки  
  
 //Вычитание дробей  
 System.*out*.print*(*"\nВычитание дробей: "*)*;//название арифметической операции  
 f.displayFraction*()*;//вывод первой дроби на экран  
 System.*out*.print*(*" - "*)*;  
 f1.displayFraction*()*;//вывод второй дроби на экран  
 System.*out*.print*(*" = "*)*;  
 //вызов метода вычитания  
 NumFraction f3 = f.subtraction*(*f1*)*;  
 f3.displayFraction*()*;//вывод результата вычитания на экран  
 System.*out*.print*(*"\nВыделение целой части: "*)*;  
 f3.fullFraction*()*;// выделение целой части дроби и вывод на экран  
 System.*out*.print*(*"\nСокращение дроби и перевод в десятичный вид: "*)*;  
 f3.simplify*()*;// сокращение дроби и вывод на экран  
 f3.decimal*()*;//перевод обыкновенной дроби в десятичную  
 System.*out*.println*()*;//перенос строки  
  
 //Умножение дробей  
 System.*out*.print*(*"\nУмножение дробей: "*)*;//название арифметической операции  
 f.displayFraction*()*;//вывод первой дроби на экран  
 System.*out*.print*(*" \* "*)*;  
 f1.displayFraction*()*;//вывод второй дроби на экран  
 System.*out*.print*(*" = "*)*;  
 //вызов метода умножения  
 NumFraction f4 = f.multiplication*(*f1*)*;  
 f4.displayFraction*()*;//вывод результата умножения на экран  
 System.*out*.print*(*"\nВыделение целой части: "*)*;  
 f4.fullFraction*()*;// выделение целой части дроби и вывод на экран  
 System.*out*.print*(*"\nСокращение дроби и перевод в десятичный вид: "*)*;  
 f4.simplify*()*;// сокращение дроби и вывод на экран  
 f4.decimal*()*;//перевод обыкновенной дроби в десятичную  
 System.*out*.println*()*;//перенос строки  
  
 //Деление дробей  
 System.*out*.print*(*"\nДеление дробей: "*)*;//название арифметической операции  
 f.displayFraction*()*;//вывод первой дроби на экран  
 System.*out*.print*(*" : "*)*;  
 f1.displayFraction*()*;//вывод второй дроби на экран  
 System.*out*.print*(*" = "*)*;  
 //вызов метода деления  
 NumFraction f5 = f.division*(*f1*)*;  
 f5.displayFraction*()*;//вывод результата деления на экран  
 System.*out*.print*(*"\nВыделение целой части: "*)*;  
 f5.fullFraction*()*;// выделение целой части дроби и вывод на экран  
 System.*out*.print*(*"\nСокращение дроби и перевод в десятичный вид: "*)*;  
 f5.simplify*()*;// сокращение дроби и вывод на экран  
 f5.decimal*()*;//перевод обыкновенной дроби в десятичную  
 System.*out*.println*()*;//перенос строки  
  
 in.close*()*;//закрываем сканер  
  
 //Вызов конструктора с параметрами по умолчанию  
 System.*out*.print*(*"\nВызываем конструктор с параметрами по умолчанию: "*)*;  
 NumFraction f6 = new NumFraction*()*;  
 f6.displayFraction*()*;//вывод конструктора с параметрами по умолчанию на экран  
 System.*out*.println*()*;//перенос строки  
  
 //Setters  
 f6.setNumerator*(*5*)*;//Задаём новое значение числителя через setters  
 f6.setDenominator*(*15*)*;//Задаём новое значение знаменателя через setters  
  
 //Getters  
 System.*out*.println*(*"Выводим новое значение числителя на экран с помощью getters: " + f6.getNumerator*())*;  
 System.*out*.println*(*"Выводим новое значение знаменателя на экран с помощью getters: " + f6.getDenominator*())*;  
  
 System.*out*.println*(*"Во всех остальных методах работает конструктор с двумя параметрами"*)*;  
 *}  
}*

Тесты:









