

+ getters, setters

+ toString, display

+ конструкторы

Код программы Main

package com.company;  
  
import java.beans.FeatureDescriptor;  
import java.util.Scanner;  
  
public class Fraction {  
  
  
  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.*in*);  
 FractionsMethods f = new FractionsMethods(); //создаем дробь  
 FractionsMethods f1 = new FractionsMethods();  
 //вводим дроби  
  
 f.input(in); // вводим данные дроб  
 f1.input(in); // вводим еще одни данные дроби  
 f.printFractions(); // выводим первые введеные данные в виде дроби  
 f1.printFractions(); // выводим вторые введеные данные в виде дроби  
 System.*out*.println('\n'); // вводим пробел между строк для лучей читабельности  
  
 //Производим математические операции с дробями  
 //1. Сложение дробей  
 System.*out*.println("Складываем дроби: ");  
 f.printFractions();  
 System.*out*.print(" + ");  
 f1.printFractions();  
 System.*out*.print(" = ");  
 FractionsMethods f3 = f.sum(f1); // обращаемся к методу сложения  
 f3.printFractions(); // выводим итог сложения дробей  
 System.*out*.println(); //ставим пробел для лучшей читаемости  
 System.*out*.print("Выделение целой части: ");  
 f3.wholeFractions(); // вывод целой части дроби  
 System.*out*.println(); //ставим пробел для лучшей читаемости  
 System.*out*.print("Сокращение дроби: ");  
 f3.reduction();  
 System.*out*.println('\n'); // вводим пробел между строк для лучей читабельности  
  
 //2. Вычитание дробей  
 System.*out*.println("Вычитаем дроби: ");  
 f.printFractions();  
 System.*out*.print(" - ");  
 f1.printFractions();  
 System.*out*.print(" = ");  
 FractionsMethods f4 = f.min(f1); // обращаемся к методу сложения  
 f4.printFractions(); // выводим итог вычитания дробей  
 System.*out*.println(); //ставим пробел для лучшей читаемости  
 System.*out*.print("Выделение целой части: ");  
 f4.wholeFractions(); // вывод целой части дроби  
 System.*out*.println(); //ставим пробел для лучшей читаемости  
 System.*out*.print("Сокращение дробей: ");  
 f4.reduction();  
 System.*out*.println('\n'); // вводим пробел между строк для лучей читабельности  
  
 //3. Умножение дробей  
 System.*out*.println("Умножение дробей: ");  
 f.printFractions();  
 System.*out*.print(" \* ");  
 f1.printFractions();  
 System.*out*.print(" = ");  
 FractionsMethods f5 = f.multiply(f1);  
 f5.printFractions();  
 System.*out*.println(); //ставим пробел для лучшей читаемости  
 System.*out*.print("Выделение целой части: ");  
 f5.wholeFractions(); // вывод целой части дроби  
 System.*out*.println(); //ставим пробел для лучшей читаемости  
 System.*out*.print("Сокращение дробей: ");  
 f5.reduction();  
 System.*out*.println(); // вводим пробел между строк для лучей читабельности  
  
 //4. Деление дробей  
 System.*out*.println("Деление дробей: ");  
 f.printFractions();  
 System.*out*.print(" / ");  
 f1.printFractions();  
 System.*out*.print(" = ");  
 FractionsMethods f6 =f.division(f1);  
 f6.printFractions();  
 System.*out*.println(); //ставим пробел для лучшей читаемости  
 System.*out*.print("Выделение целой части: ");  
 f6.wholeFractions(); // вывод целой части дроби  
 System.*out*.println(); //ставим пробел для лучшей читаемости  
 System.*out*.print("Сокращение дробей: ");  
 f6.reduction();  
 System.*out*.println('\n'); // вводим пробел между строк для лучей читабельности  
 in.close(); //закрыли сканер  
 System.*out*.println(); //ставим пробел для лучшей читаемости  
  
 //Вызов конструктора с параметрами по умолчанию  
 System.*out*.print("Вызываем конструктор с параметрами по умолчанию: ");  
 FractionsMethods f7 = new FractionsMethods();  
 f7.printFractions();  
 System.*out*.println(); //ставим пробел для лучшей читаемости  
  
 //Setters  
 f7.setA(8);//Задаём новое значение числителя через setters  
 f7.setB(12);//Задаём новое значение знаменателя через setters  
  
 //Getters  
 System.*out*.println("Выводим новое значение числителя на экран с помощью getters: " + f7.getA());  
 System.*out*.println("Выводим новое значение знаменателя на экран с помощью getters: " + f7.getB());  
  
  
 }  
}

Код программы ( методы + конструкторы + getters, setters)

package com.company;  
  
import java.util.Scanner;  
  
public class FractionsMethods {  
  
 private String name; //имя  
 private int a; // числитель  
 private int b; // знаменатель  
 private int whole = 1;  
  
 // методы класса FractionsMethods  
  
 // 1 конструктор по умолчанию - без параметров  
 public FractionsMethods() {  
 //инициализация полей  
 name = "Fraction";  
 a = 0;  
 b = 0;  
 // System.out.println("Конструктор дробей работает");  
 }  
  
 //Getters  
 public String getName() {  
 return name;  
 }  
  
 public int getA() {  
 return a;  
 }  
  
 public int getB() {  
 return b;  
 }  
  
 //Setters  
 public void setA(int value) {  
 a = value;  
 }  
  
 public void setB(int value) {  
 b = value;  
 }  
  
 //1. Конструктор с тремя параметрами  
 public FractionsMethods(String \_name, int \_a, int \_b) {  
 name = \_name;  
 a = \_a;  
 b = \_b;  
  
 }  
  
 //2. Конструктор с двумя параметрами  
 public FractionsMethods(int \_a, int \_b) {  
 a = \_a;  
 b = \_b;  
 }  
  
 //3. Еще один конструктор с тремя параметрами  
 public FractionsMethods(char \_nameLetter, int \_a, int \_b) {  
 name = "" + \_nameLetter;  
 a = \_a;  
 b = \_b;  
 }  
  
 // осуществляем вывод с клавиатуры  
 public void input(Scanner in) {  
 while (true) {  
 System.*out*.println("Вводим дроби: ");  
 System.*out*.print("Введите числитель: ");  
 a = in.nextInt();  
 System.*out*.print("Введите знаменатель: ");  
 b = in.nextInt();  
 if (a == 0 || b < 1)  
 System.*out*.println("Ввод некорректен. Введите знаменатель больше 0 ");  
 else break;  
 }  
 }  
  
 //метод инициализации  
 public void init(String \_name, int \_a, int \_b) {  
 name = \_name;  
 a = \_a;  
 b = \_b;  
  
 }  
  
 //метод вывод класса на консоль  
 public void printFractions() {  
 System.*out*.print(toString());  
 }  
  
 //метод представления класса в виде строки - перегрузка метода toString  
 @Override  
 public String toString() {  
 return "(" + a + " / " + b + ")";  
 }  
  
 // Сложение дробей: складывает 2 объекта и возвращает новый объект  
 public FractionsMethods sum(FractionsMethods summand) {  
 FractionsMethods res;  
 if (b == summand.b) { //если знаменатели равны, мы складываем числители  
 res = new FractionsMethods(name + summand.name,  
 a + summand.a, b);  
 } else { // приводим к наибольшему общему делителю  
 res = new FractionsMethods(name + summand.name,  
 (a \* summand.b) + (b \* summand.a),  
 b \* summand.b);  
 }  
 return res;  
 }  
  
 //Вычитание дробей: вычитаем 2 объекта и возвращает новый объект  
 public FractionsMethods min(FractionsMethods minus) {  
 FractionsMethods res;  
 if (b == minus.b) { //если знаменатели равны, мы вычитаем числители  
 res = new FractionsMethods(name + minus.name,  
 a - minus.a, b);  
 } else { // приводим к наибольшему общему делителю  
 res = new FractionsMethods(name + minus.name,  
 (a \* minus.b) - (b \* minus.a),  
 b \* minus.b);  
 }  
 return res;  
 }  
  
 //Умножение дробей: умножаем 2 объекта и возвращает новый объект  
 public FractionsMethods multiply(FractionsMethods mult) {  
 FractionsMethods res = new FractionsMethods(name + mult.name,  
 a \* mult.a,  
 b \* mult.b);  
 return res;  
 }  
  
 //Деление дробей: делим 2 объекта и возвращает новый объект  
 public FractionsMethods division(FractionsMethods divis) {  
 FractionsMethods res;  
 if (a > 0 && divis.a > 0) {  
 res = new FractionsMethods(name + divis.name,  
 a \* divis.b, b \* divis.a);  
 } else {  
 res = new FractionsMethods(name + divis.name,  
 a \* divis.b \* (-1),  
 b \* Math.*abs*(divis.a));  
 }  
 return res;  
 }  
  
 //Выделение целочиленной части дробного числа  
 public void wholeFractions() {  
 if (a % b != 0) {  
 whole = a / b;  
 a %= b;  
 if (whole != 0) {  
 System.*out*.print(whole + "(" + Math.*abs*(a) + "/" + b + ")");  
 } else if (a % b == 0) {  
 whole = a / b;  
 System.*out*.print(whole);  
 } else if (a > b) {  
 whole = a / b;  
 System.*out*.print(whole);  
 } else System.*out*.print("Дробь не имеет целой части!");  
 } else System.*out*.print("Дробь не имеет целой части!");  
 }  
  
  
 // Получение наибольшего общего делителя (НОД)  
 public int nod() {  
 int nod = 1;  
 for (int i = 1; i <= Math.*abs*(a) && i <= b; i++) {  
 if (a % i == 0 && b % i == 0) {  
 nod = i;  
 }  
 }  
 return nod;  
 }  
  
 // Сокращение дробей  
 public void reduction() {  
 FractionsMethods res = new FractionsMethods(Math.*abs*(a) / nod(), b / nod());  
 if (whole != 0) {  
 System.*out*.print(whole + "" + res + "");  
 } else if (a == 0) {  
 System.*out*.print(0);  
 } else {  
 res = new FractionsMethods(Math.*abs*(a) / nod(), b / nod());  
 System.*out*.print(res);  
 }  
 }  
}

Тесты программы:

С отрицательным числителями :

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Одинаковые положительные числителями и знаменателями:

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

С положительной и отрицательной дробью

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

С разными отрицательными дробями

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

При вводе некорректных данных (отрицательных и нулевых знаменателей )

Изображение выглядит как текст

Автоматически созданное описание

Разные положительные дроби

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание