```java
public class TestRationals {
    // it all starts here
    public static void main(String[] args) {
        Rational rational1, rational2;

        // define the Rationals
        rational1 = new Rational(1, 2);
        rational2 = new Rational(6, 9);
        System.out.println("First rational is: " +rational1+ " (equivalent to
" +rational1.toDouble()+ ")");
        System.out.println("Second rational is: " +rational2+ " (equivalent to
" +rational2.toDouble()+ ")");

        // do some basic math
        System.out.println();
        System.out.println("Sum: " + rational1.add(rational2));
        System.out.println("Difference: " + rational1.subtract(rational2));
        System.out.println("Product: " + rational1.multiply(rational2));
        System.out.println("Quotient: " + rational1.divide(rational2));

        // equality
        System.out.println();
        System.out.println("Are the rationals equal? "
+rational1.equals(rational2));
        System.out.println("Is the first equal to 1/2? " +rational1.equals(new
Rational(1,2)));

        // accessors & mutators
        System.out.println();
        System.out.println("The numerator of the first rational is "
+rational1.getNumerator()+ ".");
        System.out.println("The denominator of the first rational is "
+rational1.getDenominator()+ ".");
        System.out.println("Changing numerator to 6, and denominator to
7...");
        rational1.setNumerator(6);
        rational1.setDenominator(7);
        System.out.println("  ...and the result is: " +rational1);

        // fun with zero
        System.out.println();
        Rational rational0, rationalI, rationalX;
        rational0 = new Rational(0, 2);
        System.out.println("Zero as a rational is " +rational0+ " (equivalent
to " +rational0.toDouble()+ ").");
        rationalI = new Rational(-16, 0);
        System.out.println("Negative infinity as a rational is " +rationalI+ "
(equivalent to " +rationalI.toDouble()+ ").");
        rationalX = new Rational(0, 0);
        System.out.println("NaN as a rational is " +rationalX+ " (equivalent
to " +rationalX.toDouble()+ ").");
    }
}
```