

**Task2** – Базовые возможности языка:

1. Изучить интерфейс для вещественных чисел `FractionNumber`.
2. Создать класс имплементации интерфейса `FractionNumber`.
3. Изучить интерфейс класса математических операций `FractionNumberOperation`.
4. Создать имплементацию этого интерфейса.
5. Создать пример использования по каждой из функций класса имплементирующего `FractionNumberOperation`.

# Interface FractionNumber

```
public interface FractionNumber
```

Интерфейс представляющий число в виде дроби. Для вывода ошибок функции могут бросать `IllegalArgumentException` Данный интерфейс может быть использован для хранения обычных дробей, например 1/2, 1/4, 3/7 и т.д.

## Version:

1.0

## Author:

nedis

## Field Summary

static int	<a href="#"><b>DEFAULT_DIVISOR_VALUE</b></a> Значение делителя после создания обькта конструктором
------------	---

## Method Summary

int	<a href="#"><b>getDividend</b></a> ( ) Получить делимое
int	<a href="#"><b>getDivisor</b></a> ( ) Получить делитель
void	<a href="#"><b>setDividend</b></a> (int dividend) Установить делимое
void	<a href="#"><b>setDivisor</b></a> (int divisor) Установить делитель
<a href="#"><b>String</b></a>	<a href="#"><b>toString</b></a> ( ) Получить символьное представление числа Например если делимое = 1, делитель = 4, то символьное представление числа равно 1/4
double	<a href="#"><b>value</b></a> ( ) Получить значение как вещественное число.

## Field Detail

### DEFAULT\_DIVISOR\_VALUE

```
static final int DEFAULT_DIVISOR_VALUE
```

Значение делителя после создания обькта конструктором

#### See Also:

[Constant Field Values](#)

## Method Detail

### setDividend

void **setDividend**(int dividend)

Установить делимое

**Parameters:**

dividend - делимое

---

### getDividend

int **getDividend**()

Получить делимое

**Returns:**

делимое

---

### setDivisor

void **setDivisor**(int divisor)

throws [IllegalArgumentException](#)

Установить делитель

**Parameters:**

divisor - делитель

**Throws:**

[IllegalArgumentException](#) - если пытаются установить делитель равный 0

---

### getDivisor

int **getDivisor**()

Получить делитель

**Returns:**

делитель

---

### value

double **value**()

Получить значение как вещественное число. Например если делимое = 1, делитель = 4, то вещественное число для дроби  $1/4 = 0.25$

**Returns:**

вещественное число для заданной дроби

---

## toString

[String](#) toString()

Получить символьное представление числа Например если делимое = 1, делитель = 4, то символьное представление числа равно 1/4

**Overrides:**

[toString](#) in class [Object](#)

**Returns:**

символьное представление числа