

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:

alex

Field Summary

static int	DEFAULT_DIVISOR_VALUE Значение делителя после создания объекта конструктором
------------	--

Method Summary

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

Field Detail

DEFAULT_DIVISOR_VALUE

`static final int DEFAULT_DIVISOR_VALUE`

Значение делителя после создания объекта 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