Ｊａｖａプログラミング Ⅱ

９回目　例外処理　課題

確認〇×問題

次の各文は正しいか誤っているか答えなさい。

(1)　例外とは、プログラムの実行時に発生し、本来の処理の流れを妨げるエラーである
(2)　例外は、Throwable クラスまたはそのサブクラスのオブジェクトにより表現される
(3)　例外が発生した場合、本来の処理は中断される
(4)　例外が発生した後、例外に対する適切な処理が行われ、プログラムは常に強制終了する
(5)　例外処理は try 文を用いて記述する
(6)　１つの try 文で複数の例外に対する処理を記述することができる
(7)　独自の例外クラスを作成する場合、Throwable クラスまたはそのサブクラスを継承すればよい
(8)　例外を発生させるにはキーワード throws を使用する

課題１　次のコードを実行すると、配列要素を確保して配列変数に代入する部分（７行目）「ary = new int[-5]; 」で例外 NegativeArraySizeException が発生する。この例外は、負のサイズを持った配列要素を作成しようとした場合に発生する。try 文を用いてこの例外に対する例外処理を記述しなさい。下の例では、例外処理でメッセージを出力している。

```
class Assignment9_1
{
    public static void main(String[] args)
    {
        int[] ary;
        // 例外 NegativeArraySizeException が発生
        ary=new int[-5];
    }
}
```

（例外処理前の実行結果）

```
>java Assignment9_1
Exception in thread "main" java.lang.NegativeArraySizeException
        at Assignment9_1.main(Assignment9_1.java:7)
 -- Press any key to exit (Input "c" to continue) --
```

（例外処理後の実行例）

```
>java Assignment9_1
配列要素数の指定は負です
 -- Press any key to exit (Input "c" to continue) --
```

課題２　次はキーボードから整数を一つ読込むコードである。

① Integer.paeseInt()メソッドは文字列が表す整数値を int 型に変換する。もし、整数値以外が入力されると NumberFormatException（Unchecked 例外クラス）という例外をスローする。整数値以外が入力された場合、この例外をキャッチしてメッセージ「整数値ではありません」を表示しなさい。

※readLine()メソッドは IOException（Checked 例外クラス）という例外をスローする。この例外が発生した場合はこれまで通り

public static void main(String[] args) throws IOException

として、呼び出し元メソッドである main()メソッドに例外処理を任せる。

② 整数が入力されるまで繰り返すコードに変更しなさい。

```
import java.io.*;
class Assignment9_2
{
	public static void main(String[] args) throws IOException
	{
		BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

		int num=0;
		String str;

		System.out.println("整数を入力してください。");
		str=br.readLine();
		num=Integer.parseInt(str);
		System.out.println("入力された整数は"+num+"です。");
	}
}
```

課題３　次は除算を管理するクラス Div の宣言である。整数 i と j を受け取り i / j を戻り値とする。このとき、j がゼロであった場合には除算はできない。例外クラス Exception を継承して各自このエラーを表すクラス（クラス名は各自で決めること）を宣言しなさい。さらに、j がゼロであった場合にはこの例外をスローするようにクラス Div の宣言を変更しなさい。

```
class Div
{
	public static double calc(int i, int j)
	{
		double ans;
		ans=(double)i/j;
		return ans;
	}
}
```

課題４　課題２で作成した例外処理をもつ整数値のキーボード入力により２つの整数を入力して､除算を計算するクラス Div で除算を計算するコードをメインメソッドに記述しなさい。課題３で宣言したゼロでの除算を表す例外が発生しない場合は除算の結果を表示し、例外が発生した場合はこれをキャッチして無限大「∞」を表示するようにすること。

（実行例１）
>java Assignment9_4
整数 i ÷整数 j を計算します。
整数値 i を入力してください。
4r
整数値ではありません。
もう一度入力してください。
4
整数値 j を入力してください。
2
4 / 2 = 2.0
 -- Press any key to exit (Input "c" to continue) –

（実行例２）
>java Assignment9_4
整数 i ÷整数 j を計算します。
整数値 i を入力してください。
4
整数値 j を入力してください。
0
4 / 0 = ∞
 -- Press any key to exit (Input "c" to continue) --