



Tecnicatura Superior en Programación
Laboratorio III
Magni Gerardo

Trabajo Practico Nº 2

EXCEPCIONES

EJERCICIO 1: DIVISIÓN POR CERO

Escribe una clase llamada *DividePorCero* que pruebe a dividir un entero entre otro recogiendo la excepción de división por cero y mostrando el mensaje "Error: división por cero" si se produce.

EJERCICIO 2: CLASE FRACCIÓN

Diseña e implementa una clase *Fraccion* que permita crear fracciones (numerador y denominador enteros), con métodos para sumar, restar, multiplicar y dividirlos.

Crea una excepción *FraccionException* (declárala como una excepción explícita) que se lance siempre que en una operación de fracción se produzca un error, a saber:

- Creación de una fracción con denominador cero.
- Si el resultado de una división es una fracción con denominador cero (se dará si alguna de las fracciones a dividir tiene como numerador un cero).

Hacer que las operaciones críticas lancen esta excepción si se da el caso, con un mensaje indicativo del tipo de error (mensaje incluido en la excepción, no salida a pantalla!).

Construir un programa que pruebe el funcionamiento de la clase Fracción y sus operaciones.

EJERCICIO 3: SUMA DE NÚMEROS

Construir un programa que reciba una serie de números (enteros) introducidos como parámetros desde la línea de comandos, los sume y visualice el resultado de la suma. En caso de que alguno de los números introducidos no sea un entero, el programa lo ignorará, mostrará un mensaje de error y continuará con la suma del resto de los números.

Ayuda: Para resolver este ejercicio puede ser interesante que hagas uso de la clase Integer.

EJERCICIO 4:FINALLY

Agregue al código del ejercicio anterior un bloque finally que imprima por consola el mensaje "FIN DEL PROGRAMA" y responda:

- ¿El bloque finally se ejecuta siempre?
- ¿Se puede poner un bloque finally sin un bloque try?
- ¿Se puede poner un bloque finally sin un bloque catch?

METODOS Y PARAMETROS

EJERCICIO 1:

Al finalizar la ejecución de este fragmento de código, ¿que valor tienen las variables objetoCuenta, prefijoValor y sufijoValor?

```
public static void Main()
{
    int objetoCuenta = 50;
    incrementaValor(objetoCuenta);
    int sufijoValor = objetoCuenta++;
    int prefijoValor = ++objetoCuenta;
    decremetaValor (prefijoValor);
}
void incrementaValor(int valor){
    ++valor;
}
void decremetaValor(int valor){
    --valor;
}
```

EJERCICIO 2:

Reescriba el siguiente bloque de código usando for each.

```
public static void Main( )
{
    int[] numero = new int[10] {1,2,3,4,5,6,7,8,9,10};

    for (int i = 0; i < numero.Length; i++)
    {
        Console.WriteLine(numero[i].ToString());
    }
}
```

EJERCICIO 3:

Si la llamada a un método fuera de la siguiente forma

```
public static void Main( )
{
    CallMyMethod(6, 2, true);
}
```

Escriba la estructura del método que procese dicha llamada mostrando por consola en caso de que el tercer parámetro booleano sea true la multiplicación del primero y segundo parámetro y en caso de ser false mostrar el resultado de la división de los 2 primeros parametros.

EJERCICIO 4:

Dado el siguiente fragmento de código escriba un nuevo método aplicando sobrecarga de métodos de tal manera que reciba 2 parámetros y los concatene.

```
class Overloading
{
    static int Sumar(int a, int b)
    {
        return a + b;
    }

    // completar método sobrecargado aquí

    public static void Main( )
    {
        Console.WriteLine(Sumar(1,2));
        // llamada al método faltante

        Console.WriteLine(Sumar("HOLA", "MUNDO"));
    }
}
```

TABLAS o ARRAYS

EJERCICIO 1:

Cree un programa que permita la carga de 10 números enteros ingresados por teclado. Almacene cada uno de los números en un arreglo de elementos enteros. Finalmente recorra el arreglo y realice la suma de los elementos almacenados. Muestre el resultado de la sumatoria por consola.

EJERCICIO 2:

Cree un programa que permita ingresar por teclado dos valores enteros el primero para indicar el número de filas y el segundo para el número de columnas. Mediante los 2 valores cree una matriz de 2 dimensiones de orden filas X columnas.

Asigne a cada elemento de la matriz el valor equivalente a la suma de la fila y la columna en que se encuentre por ejemplo para el caso de la fila 1 columna 3 el valor del elemento será 4.

Muestre por consola la matriz resultante.