I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu s/n ☎ 965936520
03008 ALICANTE
www.iesmarenostrum.com
correo@iesmarenostrum.com

Práctica 8.1. Refactorización

Crear un proyecto nuevo llamado Fecha con las clases Program y Fecha cuyo código se adjunta. El alumno deberá refactorizar el código proporcionado y documentar los cambios realizados.

Para cada cambio se deberá indicar:

1. La refactorización realizada (p. ej Cambiar nombre).
2. El motivo de dicha refactorización (p. ej. nombre de la variable x poco descriptivo).
3. Los cambios realizados en el código (código antes y después de la refactorización).

Se deberá entregar una memoria (en formato pdf) a la que se adjuntará el código definitivo de las 2 clases (en formato cs), todo comprimido en un archivo zip.

Notas sobre el código

Comentarios TODO

Sirven para marcar en el código algo "por hacer" (to do en inglés). Todos los comentarios TODO aparecen en la ventana Lista de Tareas.

```
//TODO validar los valores introducidos
```

Constructores

Son métodos que tienen el mismo nombre que la clase, no tienen tipo y que se usan principalmente para inicializar los campos de la clase. Pueden tener varios

I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu s/n ☎ 965936520
03008 ALICANTE
www.iesmarenostrum.com
correo@iesmarenostrum.com

parámetros o ninguno. En caso de no indicarse ningún constructor, se usa el constructor por defecto, que no tiene parámetros ni código.

En la clase Fecha se usan 2 constructores, uno sin parámetros (Fecha()), en el que se asignan los valores por defecto a los campos, y otro con 3 parámetros (Fecha(mes, anyo, dia)) en el que se comprueban y asignan los valores introducidos a los campos.

Palabra reservada *this*

Sirve para diferenciar los campos de la clase cuando se llaman igual que los parámetros. Normalmente se usa en constructores. Por ejemplo, en la clase:

```
public class Alumno
{
      private string nombre;
      public Alumno(string nombre)
      {
            this.nombre = nombre;
      }

}
```

`this.nombre` de la línea 6 se refiere al campo privado de la clase, mientras que `nombre` en la misma línea es el parámetro del constructor. En este caso el uso de `this` es necesario ya que, de lo contrario, la instrucción `nombre = nombre` asignaría su propio valor al parámetro introducido y no al campo, por lo que se perdería el dato.

I. E. S. MARE NOSTRUM
Beato Fco. Castelló Aleu s/n ☎ 965936520
03008 ALICANTE
www.iesmarenostrum.com
correo@iesmarenostrum.com

Palabra reservada *override*

Sirve para sobreescribir un método definido en una clase superior, en este caso el método `ToString` de la clase `Object`. La sobreescritura está relacionada con la herencia, tal y como se vió en el tema correspondiente.