



Ejercicio 1: El imperio quiere administrar todas sus fuerzas imperiales destinadas en el planeta helado de Hoth. Los programadores destinados a tal efecto tienen la misión de gestionar todos los cuerpos militares. Se plantean en primer lugar gestionar las fuerzas de tierra, cuyo grueso del ejército son los soldados... Desarrollar un primer ejercicio que permita crear dos tipos de soldados imperiales, uno el soldado de asalto estándar y el otro que será el soldado sombra. Ambos heredan de la superclase `SoldadoImperial`.

```
public class SoldadoImperial
```

```
{
    private string StrNombre;
    private string StrCodigo;
    private string TipoRifle;
```

```
// Método constructor por defecto
// Se define pero puede que no se use
public SoldadoImperial()
{
```

```
    StrNombre = "";
    StrCodigo = "";
```

```
}
```

```
// Sobrecarga del método constructor
```

```
public SoldadoImperial(string nombre, string codigo, string rifle)
```

```
{
    StrNombre = nombre;
    StrCodigo = codigo;
    TipoRifle = rifle;
```

```
}
```

```
public string NombreSoldado
```

```
{
    get { return NombreSoldado; }
    set { NombreSoldado = value.ToUpper(); }
```

```
}
```

```
public void Imprime()
```

```
{
    Console.WriteLine("SOLDADO REGISTRADO: " + this.StrNombre+ " CON CÓDIGO " +
this.StrCodigo + " CON ARMAMENTO->" + this.TipoRifle);
}
```

```
public class SoldadoSombra : SoldadoImperial
```

```
{
```

```
    private int NumRebeldes;
    public SoldadoSombra() : base()
```

```
{
```

```
        NumRebeldes = 0;
```

```
}
```

```
// Una primer manera de constructor
```

```
/*public SoldadoSombra(string nombre, string codigo, int Rebeldesmuertos) {
```

```
    StrNombre = nombre;
```

```
    StrCodigo = codigo;
```

```
    NumRebeldes = Rebeldesmuertos;
```

```
*/
```

```
//Una segunda manera invocando a la clase padre
```

```
public SoldadoSombra(string nombre, string codigo, string TipoRifle, int Rebeldesmuertos)
: base(nombre, codigo, TipoRifle)
```

```
{
```

```
    NumRebeldes = Rebeldesmuertos;
```

```
}
```

```
public void ImprimeRebeldes()
```

```
{
```

```
    base.Imprime();
```

```
    Console.WriteLine("HA FINIQUITADO " + this.NumRebeldes.ToString()+" SUCIOS REBELDES..");
```

```
}
```

```
}
```

```
class Program
```

```
{
```

```
    static void Main(string[] args)
```

```
{
```

```
        SoldadoSombra Soldado1 = new SoldadoSombra( FLIN , FN-2199 , BLASTER , 0);
```

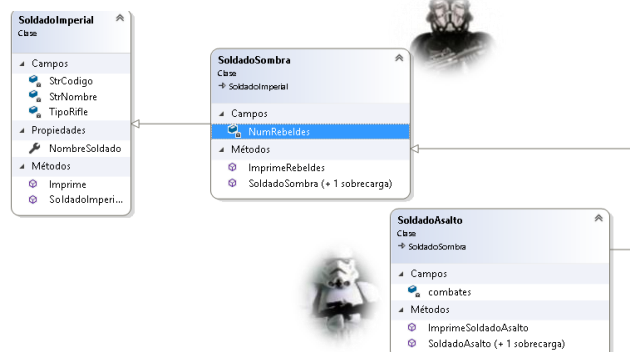
```
        Soldado1.ImprimeRebeldes();
```

```
        Console.ReadLine();
```

```
}
```

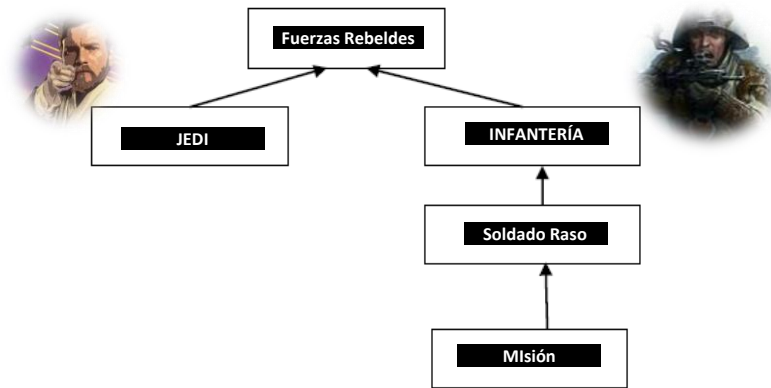
```
}
```

Ejercicio 2: El ejército imperial tiene otra jerarquía de tropas de asalto, son los soldados de Asalto los cuales se subordinan a los soldados sombra. Crea una nueva clase con herencia del soldado Sombra como se muestra. Al cargar al soldado de asalto y al introducir el número de combates, si es superior a 100 asciende a categoría sombra Mensaje correspondiente).





Ejercicio 3: La alianza rebelde te necesita.. Se avecina el ataque final ala estrella de la muerte y debemos saber con que tropas contamos. Luck Skywalker también se apunta a la fiesta por tanto debemos gestionar tropa y Jedi's conjuntamente. Se nos plantea el siguiente escenario:



- Para la superclase Fuerzas rebeldes se deben definir nombre (terrestres, Aéreas, o Jedi). La subclase jedi debe mostrarnos toda la información referente al caballero, su nombre, si es Padawan, o Maestro y el color de su sable laser).
- Para la clase Infantería considerar los atributos: Escuadra (código de tipo String), nivel de experiencia en combate).
- Para la clase Soldado Raso considerar los atributos: nombre del soldado, planeta de origen, especialidad (enumeración dividida en fusiler o, zapador, pilotoRebelde).
- Para la clase Misión considerar los atributos: nombre de la misión, total de horas estimadas, número de horas completadas.
- La solución se debe manejar a través de un Winforms (WPF) o de consola, es indistinto, pero ha de permitir las siguientes acciones:

a) Crear los objetos de tipo Misión, solicitando los datos al usuario.

b) Verificar y mostrar cada escuadra su número de misión y las horas tanto estimadas como totales para desactivar el escudo térmico que rodea la estrella de la muerte (es decir, todos los datos necesarios).

Ejercicio 4: Eres de la alianza o eres rebelde, demuéstalo en este último ejercicio. Se desea crear un formulario de altas bajas y modificaciones. En primer lugar crearemos una base de datos llamada BatallaEndor, y una tabla denominada Naves en donde se deberán almacenar todas las fuerzas disponibles para este último ataque.

| ID NAVE | TIPO DE NAVE | BANDO |
|---------|------------------|----------|
| 1 | FRAGATA MÉDICA | REBELDE |
| 2 | CRUCERO IMPERIAL | IMPERIAL |

