


 <p><b>IES JUAN BOSCO</b> www.iesjuanbosco.es</p>	<p align="center"><b>I.E.S. "JUAN BOSCO"</b> <b>CFGS DESARROLLO DE APLICACIONES WEB</b> <b>PROGRAMACIÓN</b></p>	 <p align="right">Página 1 de 5</p>
	<p align="center"><b>UT2. Utilización de objetos</b></p>	

Queremos desarrollar un programa que permita realizar la gestión del campeonato de Fórmula 1, para ello debemos desarrollar las siguientes clases:



### 1. Clase **Escuderia**

- Esta clase almacenará el nombre de la escudería, la fecha de fundación, el presupuesto que dispone de un año, las victorias conseguidas durante el campeonato. Los tipos de datos de cada uno de los atributos será elección del alumno.
- Crea los siguientes métodos:
  - Constructor por defecto. Inicializa los atributos a los siguientes valores:
    - Nombre: "Escudería por defecto".
    - Fecha Fundación: 01/01/2000
    - Presupuesto: 0.00
    - Victorias: 0
  - Constructor con parámetros donde inicializa cada uno de los atributos de la clase.
  - Métodos tanto para consultar como para modificar cada uno de los atributos de la clase.
  - Método denominado **quiebra()**, devolverá TRUE si el presupuesto es 0 y FALSE en caso contrario (*obligatorio, utilizar el operador condicional*).
  - Método llamado **bonus()**, NO RECIBE ningún parámetro y NO DEVUELVE NADA. Aumenta el presupuesto en un tanto por cierto del que dispone. El tanto por ciento a aumentar se generará de manera aleatoria (*el número aleatorio será entre 1 y 20*). Es decir, generará un número aleatorio entre 1 y 20, y dicho número será el tanto por ciento del presupuesto a aumentar.
  - Método **toString()**. Devuelve una cadena de texto donde se detalle el nombre de la escudería, su fecha de fundación, el presupuesto que dispone y el número de victorias acumuladas.

 <p><b>IES JUAN BOSCO</b> www.iesjuanbosco.es</p>	<p align="center"><b>I.E.S. "JUAN BOSCO"</b> <b>CFGS DESARROLLO DE APLICACIONES WEB</b> <b>PROGRAMACIÓN</b></p>	 <p align="right">Página 2 de 5</p>
	<p align="center"><b>UT2. Utilización de objetos</b></p>	



## 2. Clase **Bolido**

- Almacenará datos como pueden ser: el nombre del bolido, un objeto donde se asocie a la escudería que pertenece, el fabricante del motor, el fabricante de los neumáticos, el peso en kg, el número de caballos.
- Métodos:
  - Constructor por defecto. No contiene nada de código.
  - Constructor que únicamente recibe por parámetro un objeto de tipo escudería. El resto de atributos inicializarlos a los siguientes valores:
    - Nombre: "Coche por defecto".
    - Fabricante Motor: "Anónimo".
    - Fabricante Neumáticos: "Anónimo".
    - Peso: 650.80
    - Caballos: 700
  - Constructor con parámetros donde inicializa cada uno de los atributos de la clase.
  - Métodos tanto para consultar como para modificar cada uno de los atributos de la clase.
  - Método denominado **aumentarPotencia**, recibirá por parámetro el número de caballos que se incrementarán a los caballos actuales.
  - Método **toString()**. Devuelve una cadena de texto donde se detalle los valores de cada uno de los atributos.

 <p><b>IES JUAN BOSCO</b> www.iesjuanbosco.es</p>	<p align="center"><b>I.E.S. "JUAN BOSCO"</b> <b>CFGS DESARROLLO DE APLICACIONES WEB</b> <b>PROGRAMACIÓN</b></p>	 <p align="right">Página 3 de 5</p>
	<p align="center"><b>UT2. Utilización de objetos</b></p>	



### 3. Clase **Piloto**

- Guardará el nombre del piloto, la nacionalidad, un objeto que le vincule a una escudería, otro objeto asociado al bólido y el número de puntos que acumula durante el campeonato.
- Métodos:
  - Constructor por defecto, no contiene nada de código.
  - Constructor donde recibe únicamente los objetos asociados al bólido y la escudería (*el resto de atributos quedan sin inicializarse*)
  - Constructor con parámetros donde inicializa cada uno de los atributos de la clase.
  - Métodos tanto para consultar como para modificar cada uno de los atributos de la clase.
  - Método denominado **haPuntuado**, que devuelve TRUE si dispone de más de 0 puntos y FALSE en caso contrario (*utilizar el operador condicional*).
  - Método **toString()**. Devuelve una cadena de texto donde se detalle el nombre del piloto, la nacionalidad, el nombre de la escudería a la que pertenece, el nombre del coche que conduce y el número de puntos acumulados.

 <p><b>IES JUAN BOSCO</b> www.iesjuanbosco.es</p>	<p align="center"><b>I.E.S. "JUAN BOSCO"</b> <b>CFGS DESARROLLO DE APLICACIONES WEB</b> <b>PROGRAMACIÓN</b></p>	 <p align="center">Página 4 de 5</p>
	<p align="center"><b>UT2. Utilización de objetos</b></p>	

#### 4. Clase **Carrera**

- Almacena datos como el nombre de la carrera, el nombre del circuito, y 3 objetos de tipo piloto donde se guardará el campeón, subcampeón y el piloto que finalizó en tercera posición.
- Métodos
  - Constructor por defecto, no contiene nada de código.
  - Constructor con parámetros donde inicializa cada uno de los atributos de la clase.
  - Métodos para consultar cada uno de los atributos de la clase.
  - Métodos para modificar los atributos nombre y circuito de la carrera.
  - Método denominado **setCampeon**. Recibirá como parámetro el objeto piloto que ha sido campeón de la carrera (*por tanto, deberá modificar el atributo de la clase que almacena el objeto piloto campeón de la carrera*), además incrementará en **25 puntos** los puntos del piloto y sumará una victoria más a la escudería de la cual pertenece el piloto.
  - Método **setSubcampeon**. Igual que el método anterior, pero en este caso recibirá el objeto que ha sido subcampeón de la carrera, y añadirá **15 puntos** en los puntos totales del piloto. Además, para que en futuras carreras pueda optar a la victoria, este método aumentará en 50 caballos la potencia del bólido del piloto.
  - Método **set3posicion**. Como los métodos anteriores, recibirá el objeto del piloto que ha quedado en tercera posición y el sumará además **5 puntos**. Con el fin de compensar el nivel y aproximarse al ganador, se deberá bonificar a la escudería de la que pertenece el piloto con un aumento de su presupuesto.

 <p><b>IES JUAN BOSCO</b> www.iesjuanbosco.es</p>	<p align="center"><b>I.E.S. "JUAN BOSCO"</b> <b>CFGS DESARROLLO DE APLICACIONES WEB</b> <b>PROGRAMACIÓN</b></p>	 <p align="right">Página 5 de 5</p>
	<p align="center"><b>UT2. Utilización de objetos</b></p>	

### 5. Clase **Principal**

Contendrá el método principal de la aplicación (*main*). Realizar las siguientes acciones:

- Crea 3 escuderías, 3 bólidos y 5 pilotos.
- Muestra la información de los 5 pilotos y las escuderías.
- Crea 2 objetos de tipo Carrera y asigna los campeones, subcampeones y los que han finalizado en tercera posición.
- Vuelve a mostrar la información de los 5 pilotos y las escuderías y comprueba que los datos se han actualizado, dependiendo de la posición que ocupó cada piloto.