

0485.- Programación.

[Área personal](#) / [Mis cursos](#) / [Programación](#) / [U5 - POO. Herencia. Interfaces.](#) / [U5 - POO. Herencia. Interfaces 20/21 Mañana](#)

U5 - POO. Herencia. Interfaces 20/21 Mañana

EJERCICIO 1

La empresa ATRIOS ha recibido un encargo por parte del CDI(Comité Deportivo Internacional) para crear una aplicación que permita gestionar las ediciones de Juegos Deportivos Mundiales.

Para cada edición de los Juegos debemos guardar la siguiente información:

- El año de celebración. Debemos asegurarnos de que es múltiplo de 4.Si no es así debes decidir qué hacer.
- La sede en la que se van a celebrar.
- Una lista de países participantes.Deberé poder añadir y quitar países a dicha lista.
- Una lista de deportes cuyas competiciones va a albergar dicha edición de los Juegos Deportivos.Deberé añadir y quitar deportes a dicha lista.

De cada País deberemos conocer la siguiente información:

- El nombre del país.
- El número de participantes que aportan a las competiciones.

La clase Deporte no es instanciable pero recoge una serie de datos que son comunes a todos los deportes. A saber:

- Nombre del deporte
- Nombre del pabellón en el que se a celebrar.
- La lista de los participantes inscritos. Deberé poder añadir y quitar participantes de dicha lista.

Como es año covid se van a celebrar pocos deportes Atletismo, Judo y Baloncesto. Del Atletismo guardaremos el número de pruebas , del Judo el número de tatamis que se van a habilitar y del Baloncesto el número de equipos del torneo.

La clase Participante no podrá instanciarse pero contendrá lo siguiente:

- El nombre del atleta.
- La edad del atleta.
- Y un método que deben tener todas las clases hijas que se llame hacerjuramento().

Tenemos por lo tanto tres tipos de Participantes:

- Los Atletas, que cuando realicen el juramente mostrarán por pantalla, "Yo XXXXXX, como atleta, juro los valores deportivos mundiales". Además debemos guardar el nombre de la prueba en la que participan.
- Los Judokas, que cuando realicen el juramento mostrarán por pantalla, "Yo XXXXXX, como judoka, juro los valores deportivos mundiales". Además deberemos guardar el peso de su categoría (un número entero).
- Los jugadores de baloncesto, que cuando realicen en juramento mostrarán por, "Yo XXXXXX, como jugador de baloncesto, juro los valores deportivos mundiales". Además deberemos guardar su altura (un número con decimales).

XXXXXX será sustituido por el nombre del participante.

Adicionalmente los participantes tienen comportamientos diferentes, los atletas podrán correr(), mostrando por pantalla "Voy a correr", los judokas podrán luchar(), mostrando por pantalla "Voy a pelear" y los jugadores de baloncesto podrán encestar() mostrando por pantalla "2 puntos!!!!". Estos comportamientos deberán poder añadirse participantes de otros deportes en un futuro. Y un deporte podría tener más de uno de esos comportamientos.

Debemos también implementar los siguientes métodos donde sea necesario:

- mostrarPaíses() que mostrará los países ordenador de manera decreciente por el número de participantes que aportan.
- mostrarPaísesParaCeremonia() que mostrará la lista de los países ordenados por su nombre. Será el orden por defecto que queremos establecer.
- mostrarParticipantes() que mostrará los participantes de un determinado deporte ordenados por edad.
- mostrarJugadoresBaloncestoPorAltura() que mostrará ordenador por altura una serie de participantes de Baloncesto.

EN EL MAIN DEBERÁS CREAR LOS OBJETOS NECESARIOS PARA PROBAR TODAS LAS OPERACIONES DE LAS CLASES (Excluyendo getters y setters).



Ninguno de los atributos de las clases deberá ser público y deberemos crear los métodos adicionales que consideremos necesarios.

Instrucciones para la Entrega

Entregar un **archivo comprimido (zip o rar)** que contenga cada uno de los [ficheros](#) correspondientes a los ejercicios (EjX.java siendo X el número del ejercicio)

EL fichero comprimido se llamará **Apellido1_Apellido2_U5_T1_Entrega.zip** (o rar)

Evita ñ y acentos en el nombre.

Deberán subirse a [gitHub](#) los [ficheros](#) en la carpeta correspondiente (U5_T1) que se encontrará dentro de la carpeta U5

NOTA: Para cada uno de los ejercicios se bajará un 20% del mismo por cada error sintáctico corregido por el profesor para que el ejercicio pudiera funcionar correctamente. Se tendrá también en cuenta la eficiencia de los algoritmos.

Estado de la entrega

Estado de la entrega	No entregado
Estado de la calificación	Sin calificar
Fecha de entrega	jueves, 11 de febrero de 2021, 14:35
Tiempo restante	19 minutos 12 segundos

Criterio de calificación

Optimización del código	Muchas partes del código están sin optimizar 0 puntos	Algunos detalles no están optimizados 5 puntos	El código está totalmente optimizado 10 puntos
Vehículos (abstracta)	No declara la clase como abstracta 0 puntos	Declara la clase como abstracta 5 puntos	
Vehículos - Atributos	No todos los atributos están definidos 0 puntos	Define correctamente todos los atributos 5 puntos	
Vehículos - Constructor	No establece correctamente un constructor para la clase abstracta 0 puntos	Establece correctamente un constructor para la clase abstracta 5 puntos	
Vehículos - Métodos (transportar)	Más de un método de la clase no está correctamente definido. 0 puntos	Uno de los métodos de la clase no está correctamente definido 2 puntos	Define correctamente todos los métodos necesarios para la clase 5 puntos
Terrestre - Herencia y Clase Abstracta	Ni hereda correctamente ni es abstracta 0 puntos	O no hereda correctamente o no es declarada como abstracta 2 puntos	Hereda de vehículo y es declarada abstracta 4 puntos