

Programación

EXAMEN PRÁCTICO TEMA 5 – POO

(10/02/2025)



LEE ATENTAMENTE LAS SIGUIENTES INSTRUCCIONES ANTES DE EMPEZAR:



- **Recopila en un documento de texto las evidencias de todo el examen.** Guárdalo de vez en cuando para no perder el avance de tu trabajo.
- Cuando termines, **pásalo a PDF y sube el documento creado a la entrega de AULES.**

PARTE 1: Configuración del entorno (0,5p)

1. Crea un nuevo repositorio llamado “EXAMEN_UD5_[nombre]” desde *SourceTree*. El repositorio debe crearse en local y tener su espejo en remoto, por lo tanto, sincronízalo con *GitHub*.

Pega a continuación la URL a tu nuevo repositorio de *GitHub*:

2. Crea un nuevo proyecto *Java* (*Maven*) con *IntelliJ* -o el IDE que utilices- dentro del repositorio que acabas de crear. Llámalo “EXAMEN UD5”.
3. Crea en el proyecto un paquete nuevo llamado “*carrera_solidaria*”.

Sincroniza los cambios en tu repositorio remoto.

PARTE 2: Resolución de problemas

Programa en *Java* la solución a los siguientes ejercicios. Usa el proyecto que te acabas de crear en el apartado anterior. **Si no has conseguido crearlo correctamente, utiliza alguno de los proyectos que ya tenías para los ejercicios de clase y pega la URL de *GitHub* del repositorio al que vas a subir los cambios.**

1. (6,5p) Carrera solidaria del IES MUTXAMEL por la DANA de Valencia

Dentro de unas semanas, el IES MUTXAMEL organizará una carrera solidaria para sacar fondos y enviarlos a los afectados por la DANA ocurrida en Valencia a finales del mes de octubre.

El formato que ha pensado la dirección del centro es el siguiente:

- Cada estudiante que quiera participar como corredor deberá **buscar patrocinadores** (valen profesores, familiares, amigos,...) cualquiera que quiera ser solidario.



- Cada corredor acuerda con sus patrocinadores una cantidad a pagar (€) por vuelta completada. Es decir, si se acuerda una cantidad de 1€ y el corredor da 10 vueltas al circuito, el patrocinador deberá pagar 10€ al corredor, el cual deberá guardarse el dinero recaudado para entregar su donación al centro más tarde.

Para hacer posible todo esto, el centro necesita una aplicación para poder digitalizar la inscripción de los participantes en la carrera y así tenerlo todo controlado.

1. (4,5p) En la app *KmSolidariosIESMutxamel* habrá:

- Una clase **Organizador**, con los siguientes atributos:
 - o *nombre (String)* – debe ser siempre “IES MUTXAMEL”.
 - o *edicion (String)* – se empieza por la primera (I), si se crea otra la II,...
 - o *listaCorredores (ArrayList<Corredor>)* - por temas logísticos, se limita a 700 participantes como máximo.

Por defecto, se crea sin corredores y se van agregando conforme van realizando su inscripción en conserjería.

- Una clase **Corredor**, con los siguientes atributos:
 - o *nombre (String)*
 - o *dorsal (int)* – autogenerado por orden de inscripción empezando por 1.
 - o *listaPatrocinadores (ArrayList<String>)* – se debe permitir insertar varios elementos a la vez.
 - o *importe_vuelta (double)* – si no se indica, **0.50€** por defecto.
 - o *vueltas_completadas (int)* – valor **0** por defecto hasta que se realice la carrera y se actualice.

La lista de patrocinadores se creará vacía en el momento de hacer la inscripción del corredor, y se irá actualizando conforme se vayan consiguiendo acuerdos.

A tener en cuenta...

- Dado que el instituto no quiere borrar a los corredores inscritos de su aplicación por si se realizan más ediciones de la carrera y quieren volver a participar, cualquier objeto de tipo *Organizador* y *Corredor* deben existir indistintamente dentro de la *app*.
- **No uses anotaciones de la librería *Lombok*.**

2. (2p) Funcionalidades extras a implementar en la clase *Organizador*.

- a) (1p) Implementa una funcionalidad en la app para que imprima el número total de corredores inscritos y sus nombres con el siguiente formato **[dorsal]. nombre:**

```
Corredores inscritos: 2
[1]. Luis
[2]. Carla
```

- b) (1p) Implementa una funcionalidad para que se pueda calcular el importe de la donación a realizar por un corredor en cuanto acabe la carrera y se hayan actualizado las vueltas dadas al circuito por cada uno de ellos.

Programa *main* de ejemplo

```
public class AppCarrera {

    public static void main (String[] args){

        Organizador iesmutxamel = new Organizador("I");
        System.out.println(iesmutxamel);

        Corredor corredor1 = new Corredor("Luis",1.5);
        System.out.println(corredor1);
        iesmutxamel.inscribir_corredor(corredor1);

        corredor1.insertarPatrocinador("Madre","Profe_mates","Profe_ingles");
        corredor1.insertarPatrocinador("Profe_cast");
        System.out.println(corredor1);

        Corredor corredor2 = new Corredor("Carla",0.0);
        System.out.println(corredor2);
        iesmutxamel.inscribir_corredor(corredor2);

        System.out.println(iesmutxamel);

        //a)
        iesmutxamel.mostrarCorredores();

        //b)
        corredor1.setNumero_vueltas(15);
        iesmutxamel.calcularDonacion(corredor1);

    }

}
```

Salida de ejemplo

```
"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.2\lib\idea_rt.jar=55728:C:\Progrn
- Organizador [nombre='IES MUTXAMEL', edicion='I', listaCorredores=[]]
- Corredor [nombre='Luis', dorsal=1, listaPatrocinadores=[], importe_vuelta=1.5, numero_vueltas=0]
* Nueva inscripción realizada. Corredor : Luis con dorsal 1
* Nuevos patrocinadores añadidos para el corredor Luis
* Nuevos patrocinadores añadidos para el corredor Luis
- Corredor [nombre='Luis', dorsal=1, listaPatrocinadores=[Madre, Profe_mates, Profe_ingles, Profe_cast], importe_vuelta=1.5, numero_vueltas=0]
- Corredor [nombre='Carla', dorsal=2, listaPatrocinadores=[], importe_vuelta=0.5, numero_vueltas=0]
* Nueva inscripción realizada. Corredor : Carla con dorsal 2
- Organizador [nombre='IES MUTXAMEL', edicion='I', listaCorredores=[- Corredor [nombre='Luis', dorsal=1, listaPatrocinadores=[Madre, Profe_mates, Profe_in
Corredores inscritos: 2
    [1]. Luis
    [2]. Carla
El corredor Luis ha conseguido una donación de 22.5€

Process finished with exit code 0
```

```
Profe_cast], importe_vuelta=1.5, numero_vueltas=0], - Corredor [nombre='Carla', dorsal=2, listaPatrocinadores=[], importe_vuelta=0.5, numero_vueltas=0]]]
```