

PAC de Desarrollo UF3

CFGS DAW

Módulo 03B: Programación B

1S 2024/2025



1. Normativa

Requisitos necesarios que debe cumplir el trabajo:

- La PAC de Desarrollo se debe entregar únicamente a través la tarea habilitada para tal efecto en el Campus, y dentro de los plazos (día y hora máximo) establecidos en la guía didáctica. **En caso de no cumplir dichos plazos, no se podrá enviar de forma posterior. No se evaluarán entregas por mensajes o comentarios.**
- **Solo se permitirá una entrega por alumno**, por lo que tendrás que asegurarte muy bien de que entregas el fichero correcto, de que su contenido es válido y de que lo has adjuntado a la tarea antes de finalizar el proceso de entrega.
- Para realizar esta actividad se deben consultar los contenidos del material didáctico y hacer uso de fuentes bibliográficas y recursos web.
- **Cualquier indicio de posibilidad de plagio supondrá un suspenso (0) para todas las partes implicadas**, independientemente de la fecha de entrega. Siendo conscientes de la probabilidad de que dos programas sean parecidos, si se detecta algún caso claro de plagio (como un nombre de autor erróneo), se calificará con un 0.
- **La entrega del archivo en la plataforma es responsabilidad del alumno/a**, debe asegurarse de haber subido en la plataforma el fichero correcto, ya que **en ningún caso el profesor revisará el estado de la entrega o el contenido del archivo antes del periodo de corrección**.
- El programa debe realizarse con **java exclusivamente**, en **ningún otro lenguaje de programación ni empleando ningún tipo de framework o plugin**.
- La calificación oscilará entre 0 y 10. Si no se entrega una PAC de desarrollo, la calificación equivaldrá a un 0.
- **El/La alumno/a es conocedor de esta normativa en el momento de la entrega de la PAC de desarrollo.**

1.1 NORMAS DE OBLIGADO CUMPLIMIENTO PARA ENTREGAR

- En la tarea se contestarán las preguntas pertinentes.
- En el último apartado de la tarea se entregará un fichero `Animal.java` que contenga la clase objeto de esta PAC de Desarrollo.
 - NO se entregará `Main.java`.
 - Tampoco el proyecto eclipse completo.
 - No habrá ningún directorio ni fichero comprimido, se entregará solamente el fichero `Animal.java` (sin modificar en ningún momento dicho nombre).

Incumplir cualquiera de estos puntos supondrá suspender la PAC (0).

Estas restricciones derivan de la programación de un script que descomprime todas las entregas de la PAC y las compila, preparándolas para su ejecución. Si se incumple alguna de ellas, el script será incapaz de descomprimir y compilar el programa en cuestión, y directamente se calificará como suspenso.

2. Enunciado de la actividad

Vas a crear un juego de azar donde van a competir en una carrera varios animales. Para ello vamos a tener una clase `Animal` con las siguientes características:

- Un atributo `String` para el nombre
- Un atributo `int` para la distancia recorrida durante la carrera

Y en cuanto al comportamiento de `Animal`, habrá:

- Un constructor que recibirá el nombre y que inicializará la distancia recorrida a 0. El método no mostrará nada en consola.
- Un método `correr` que recibe un valor entero (la distancia recorrida en metros). El método sumará dicha nueva cantidad de metros al atributo de distancia recorrida en los turnos anteriores. El método no mostrará nada en consola.

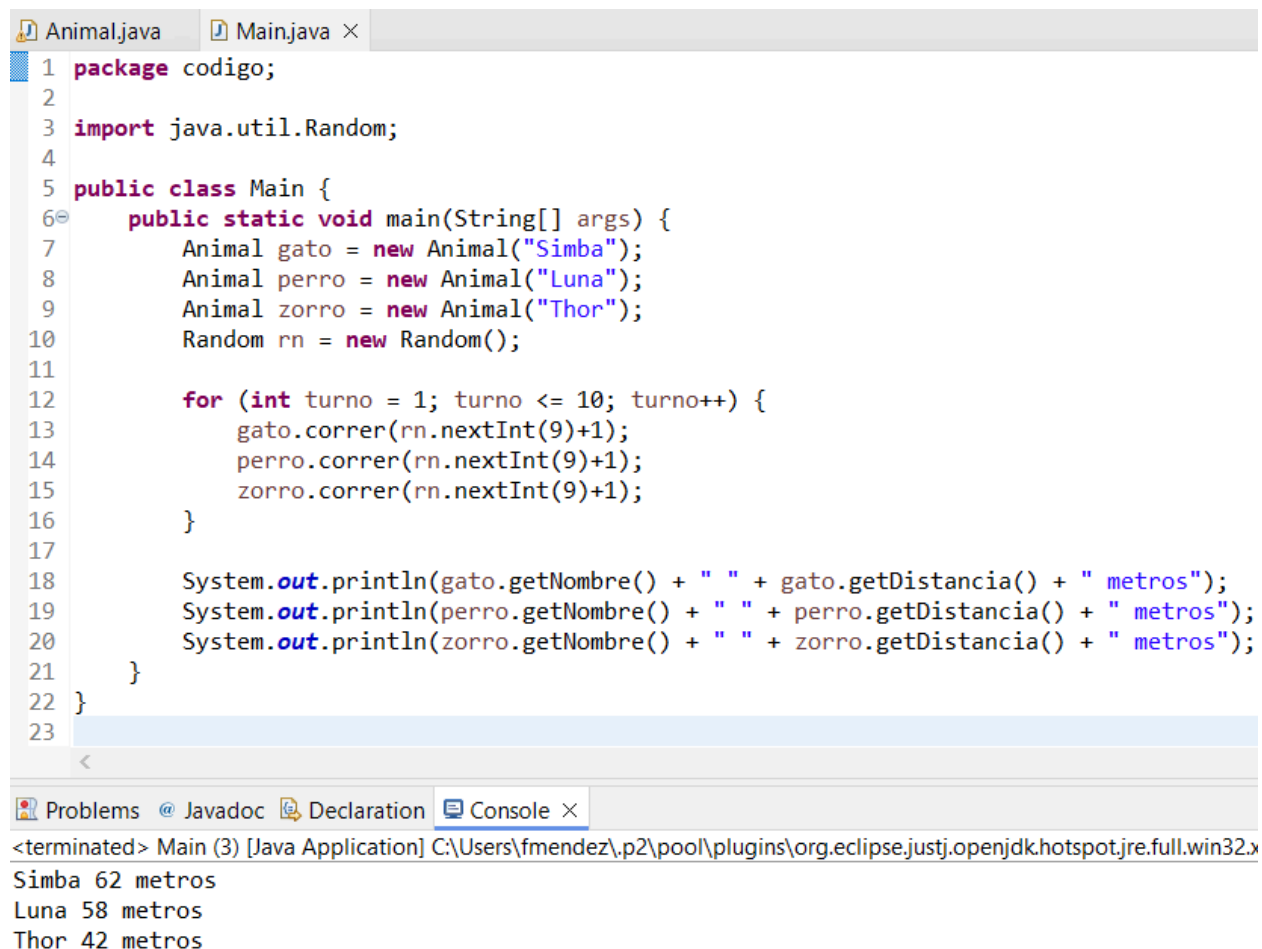
Esa clase deberá funcionar con el siguiente método `main`:

```
package codigo;
import java.util.Random;
public class Main {
    public static void main(String[] args) {
        Animal gato = new Animal("Simba");
        Animal perro = new Animal("Luna");
        Animal zorro = new Animal("Thor");
        Random rn = new Random();
        for (int turno = 1; turno <= 10; turno++) {
            gato.correr(rn.nextInt(9)+1);
            perro.correr(rn.nextInt(9)+1);
            zorro.correr(rn.nextInt(9)+1);
        }
        System.out.println(gato.getNombre() + " " + gato.getDistancia() + " metros");
        System.out.println(perro.getNombre() + " " + perro.getDistancia() + " metros");
        System.out.println(zorro.getNombre() + " " + zorro.getDistancia() + " metros");
    }
}
```

Presta especial atención a los detalles, como el nombre del paquete o los nombres de los métodos que se emplean en el método `main` para construir tu clase `Animal`. En caso de error, el programa no funcionará, suponiendo un suspenso.

2.1 EJEMPLO DE EJECUCIÓN DE MAIN.JAVA

A continuación podemos ver un ejemplo de ejecución de una carrera. Dado que los números son aleatorios, el resultado puede variar de una ejecución a otra.



```

1 package codigo;
2
3 import java.util.Random;
4
5 public class Main {
6     public static void main(String[] args) {
7         Animal gato = new Animal("Simba");
8         Animal perro = new Animal("Luna");
9         Animal zorro = new Animal("Thor");
10        Random rn = new Random();
11
12        for (int turno = 1; turno <= 10; turno++) {
13            gato.correr(rn.nextInt(9)+1);
14            perro.correr(rn.nextInt(9)+1);
15            zorro.correr(rn.nextInt(9)+1);
16        }
17
18        System.out.println(gato.getNombre() + " " + gato.getDistancia() + " metros");
19        System.out.println(perro.getNombre() + " " + perro.getDistancia() + " metros");
20        System.out.println(zorro.getNombre() + " " + zorro.getDistancia() + " metros");
21    }
22 }
23

```

Problems @ Javadoc Declaration Console ×
 <terminated> Main (3) [Java Application] C:\Users\fmendez\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x
 Simba 62 metros
 Luna 58 metros
 Thor 42 metros

3. Evaluación

La evaluación de esta actividad se realizará siguiendo la siguiente rúbrica:

1. Compilación (2 puntos). El programa compila correctamente.
2. Clase Animal (8 puntos) distribuidos de la siguiente forma:
 - a. Creación de atributos (2 puntos). Creación correcta de atributos.
 - b. Creación del método constructor (2 puntos). Método constructor correcto.
 - c. Creación de getters y setters (2 puntos). Getters y setters correctos.
 - d. Creación del método correr (2 puntos). Creación correcta del método.

Criterios	Pruebas	Terminada	Puntos
Compilación	El proyecto compila	<input type="checkbox"/>	2
Clase Animal	Creación de atributos	<input type="checkbox"/>	2
	Creación de constructor	<input type="checkbox"/>	2
	Creación de propiedades (getters y setters)	<input type="checkbox"/>	2
	Creación del método correr	<input type="checkbox"/>	2
TOTAL			