

Ex 1,2 y 3.

Como vemos aquí , hemos aplicado el ejercicio 1. Porque estamos deserializando el contenido del archivo indicado en el código.

```
C:\Users\shote\.jdk\openjdk-23.0.1\bin\java.exe
Raza: null, nombre: Dorita, Edad: 9
Raza: null, nombre: Fred, Edad: 5

Process finished with exit code 0
```

También se aplica el Ex2 porque hemos definido uno de los atributos como transient para que no de la información al deserializar.

```
public class Perro implements Serializable {

    private transient String raza; //Ejercicio 2
    private String nombre;
    private int edad;
}
```

Y si llegamos a indicar una ruta que no existiera. Pues aplicamos el ejercicio 3. Y aún seguirá adelante, sin importar que no se indique bien la ruta.

```
m(new FileOutputStream( name: "src/main/resources/Perros.ser"));
```

```
C:\Users\shote\.jdk\openjdk-23.0.1\bin\java.exe "-javaagent:D:
Algo salio mal, creando archivo con valores por defecto
Raza: null, nombre: default, Edad: 0

Process finished with exit code 0
```

Ex 4.

```
C:\Users\shote\.jdk\openjdk-23.0.1\bin\java.exe "-javaagent:D:\Descargas\...
Posicion: 1, Perro: Perro(raza=null, nombre=Dorita, edad=9)
Posicion: 2, Perro: Perro(raza=null, nombre=Manolo/, edad=3)
Posicion: 5, Perro: Perro(raza=null, nombre=Fred, edad=5)
Posicion: 8, Perro: Perro(raza=null, nombre=Junior (morci), edad=11)

Process finished with exit code 0
```

Como vemos ahora, tenemos los perros ordenados por el valor de la posición de la carrera.

Ex 5.

```
C:\Users\shote\.jdk\openjdk-23.0.1\bin\java...
Employee(name=Luis, salary=3000.0)
Boss(department=Recursos Humanos)
Boss(department=TI)

Process finished with exit code 0
```

Como se ve aquí, aunque el empleado implementa Serializable. Los atributos heredados en Boss no salen. Solo el suyo propio.