

## Enunciado

A lo largo de esta unidad has ido aprendiendo a crear tus propias clases así como sus distintos miembros (atributos y métodos). Has experimentando con la encapsulación y accesibilidad (modificadores de acceso a miembros), has creado miembros estáticos (de clase) y de instancia (de objeto), has escrito constructores para tus clases, has sobrecargado métodos y los has utilizado en pequeñas aplicaciones. También has tenido tu primer encuentro el concepto de herencia, que ya desarrollarás en unidades más avanzadas junto con otros conceptos avanzados de la Programación Orientada a Objetos.

Una vez finalizada la unidad se puede decir que tienes un dominio adecuado del lenguaje Java como para desarrollar tus propias clases y utilizarlas en una aplicación final que sea capaz de manipular un conjunto de datos simple. Dada esa premisa, esta tarea tendrá como objetivo escribir una pequeña aplicación en Java empleando algunos de los elementos que has aprendido a utilizar.

Se trata de desarrollar una aplicación Java en consola que permita gestionar trabajadores.

1. Define una clase **Funcionario** considerando(2 puntos):

- Sus *atributos* serán: nombre, apellidos, edad, fechaIngreso.
- Crea un *constructor* por defecto y otro con todos los parámetros.
- Crea métodos *get* y *set* para todos los atributos.
- Crea un método *getInfo* que devuelva toda la información en un String, de tipo: "El funcionario *nombre* *apellidos*, tiene *edad* años e ingresó en el cuerpo *fechaIngreso*."

2. Define una clase **Bombero** considerando: (3 puntos)

- Se trata de una clase *heredada* de funcionario.
- Sus *atributos* serán: puesto (de tipo Puesto) y parque (de tipo String).
- Debes definir también un atributo estático que lleve la cuenta de los *bomberos creados*.
- Crea un *constructor* por defecto y otro con todos los parámetros (utiliza el de la clase padre con la instrucción *super* y añade lo que falte).
- Crea métodos *get* y *set* para todos los atributos.
- Sobreescribe el método *getInfo* para que devuelva la información completa de Bombero.
- Crea un método *calculaAntigüedad* que al llamarlo modifique el sueldo de acuerdo a: si ha pasado un año desde el ingreso suba el sueldo un 1%, si han pasado 5 años un 5% y si han pasado 10 un 10%.

3. Define una clase **Puesto** con las siguientes indicaciones: (1,5 puntos):

- Atributos*: nombrePuesto, categoría, descripcionPuesto, sueldo.
- Crea un *constructor* que rellene puesto, categoría y descripción y establezca el sueldo en función de la categoría: peon 1500€, cabo 1800€ y especialista 2000€;
- Crea métodos *set* y *get*.

4. Crea una clase **BomberoMain** que realice lo siguiente: (3,5 puntos)

- Cree 3 objetos Puesto, uno de cada categoría, pidiendo los datos a usuario por pantalla(excepto la categoría que debe ser una de cada).
- Cree 3 objetos bombero con sus parámetros (solicitando los datos por pantalla, excepto la categoría que será uno de cada) y muestre su información por pantalla.
- Muestre el número de bomberos creados.
- Establezca la fecha de ingreso del bombero 1 a más de un año, la del bombero 2 a más de 5 años y la del bombero 3 a más de 10 años, calcule su antigüedad y muestre el sueldo nuevo por pantalla.
- Solicite al usuario el puesto del bombero1 y lo asigne al objeto, recalcule su antigüedad y muestre su sueldo por pantalla.
- Asigna al bombero 1 el Puesto del bombero 2 y al bombero 3 el Parque del bombero 1.

En general, deberías incluir **excepciones** para controlar aquellos casos en los que el uso de un método no sea posible o en el que se indique un tipo de dato erróneo.

El código fuente Java de esta clase debería incluir **comentarios** para entender su correcto funcionamiento.