

Folla 4.0. Clases con obxectos como atributos. Arrays de obxectos

1. Crea unha clase **BiciElect** que teña como atributos **nome**, **marca**, **autonomiaKms**. Define os seus getters e setters, e 2 construtores, un con parámetros e outro sen eles. Define un atributo estático **numBicis** na que gardaremos o número de bicicletas que se van creando. Define tamén un método **comparaAutonomía()**, que recibe 2 BiciElect e devolve a BiciElect con maior autonomía.

Crea agora outra clase **AppBici**, na que debes crear un array con 5 bicis. O programa deberá mostrar o menú seguinte, que debe ser xestionado cun switch:

- a) Introducir unha única bicicleta, na posición que corresponde (primeiro no 0, logo no 1, etc.). Debes controlar con **numBicis** en que posición hai que introducir a bici nova.
 - b) Mostrar todos os elementos do array. (emprega **numBicis** como índice máximo do bucle)
 - c) Cambiar os datos dunha bicicleta concreta. O usuario será preguntado polo índice do array que quere modificar e modificar esa bicicleta. Só poderá modificar unha posición na que xa exista a bicicleta.
 - d) Mostra a bicicleta con máis autonomía. Emprega o método **comparaAutonomía()** nas comparacións.
 - e) Mostra a bicicleta con menor autonomía. Emprega o método **comparaAutonomía()** nas comparacións.
 - f) Saír.
2. Crea un programa que permita xestionar as altas/baixas nunha empresa e que conste das seguintes características:

Clases: Crear dúas clases: *Empresa.java* e *Empleado.java*.

Atributos:

1.- Na clase *Empresa* deben declararse as características da empresa, tales como: *nome*, *CIF*, *localidade*, e un array dos *Empleados* pertencentes á empresa, como moito serán 6.

2.- Na clase *Empleado* deben declararse as características dun empregado que permitan identificalo dentro da empresa: *nome*, *numeroIdentificacion*, *categoría* e *soldo*.

Constructores

Empleado.

- a) Indica o nome do empregado empregado, o seu número de indentificación e a categoría á que pertence. A categoría por defecto é a de programador

Empresa.

- a) Indica o nome da empresa e o seu número máximo de empregados
- b) Indicar o nome de la empresa. O número de empregados *por defecto* será de 6.

Métodos

Empleado.

Xera automaticamente os **getters** e **setters**. Modifica os métodos que cambian ou asignan o soldo dos empregados tendo en conta a súa categoría na empresa:

programador 1100 < *soldo* < 1800

analista: $1500 < \text{soldo} < 2200$
consultor: $1700 < \text{soldo} < 3500$.
directivo: $4000 < \text{soldo} < 4800$.

Cando o saldo modificado ou asignado non está dentro dos valores asignados á súa categoría o método deberá dar un aviso e non modificar ou asignar ese soldo.

Crea un método **toString()**, que mostre todas as características dun empregado.

Empresa:

- a) Obter o *nome* da empresa.
- b) Obter o *CIF* da empresa.
- c) Obter o *número de empregados* dunha empresa.
- d) Comprobar se un empregado *pertence* á empresa.
- e) Obter o nome dun empregado a partir do seu número de identificación.
- f) Obter o número de identificación dun empregado a partir do seu nome.
- g) Amosar por pantalla o nome dos empregados da empresa.
- h) Dar de alta a un empregado na empresa
- i) Dar de baixa a un empregado na empresa empregando o número de identificación
- j) Calcular a suma do soldo dos traballadores dunha empresa.
- k) Mostra todos os empregados que ten a empresa, con todos os seus datos.
- l) Mostrar o número de empregados que pertencen a cada unha das categorías da empresa

Crea unha clase **AppEmpresa.java** que contén o código necesario para demostrar as operacións creadas:

Crea 2 empresas distintas, asignando 3 traballadores inventados a cada unha, e comproba que todos os métodos son correctos.