



Realiza los siguientes ejercicios. Cuando los termines, comprímelos en un archivo Tarea\_UF7\_tu\_nombre.

1. En un proyecto TUF7\_Ej1\_tu\_nombre implementa una clase Movil (partiendo del modelo UML que está anexo abajo). Ten en cuenta que el constructor sólo crea objetos con los atributos modelo y precio. El método llamar sacará un mensaje "Llamando..." si no estábamos hablando, en caso de estar hablando mostrará "No se puede llamar, tfno. ocupado". El método colgar mostrará un mensaje "Colgado" independientemente del valor de la variable hablando. El método mostrarMovil mostrará la información del tfno. y una frase que represente el valor de la variable hablando. Una vez creada la clase, implementa el método main dando los siguientes pasos: (3 puntos)
  - Crea un array de 4 móviles
  - Rellena el array de móviles (piensa que en este ejercicio el array es de 4 pero que podría ser de 400) pidiendo los datos por pantalla y llamando a un método insertarXPrecio(Movil [], Movil) que reciba el array de móviles y el objeto móvil a insertar. Según se van introduciendo deben ocupar las primeras posiciones del array sin dejar huecos en blanco y se tienen que ir ordenando por precio (del más barato al más caro), por lo que si entra un móvil que no es el más caro (se insertaría en la primera posición libre) se debe introducir en la posición que le corresponda y desplazar a la derecha el resto móviles más caros que él. Además el método debe devolver un true si se ha insertado correctamente o false si no ha sido así (al array estaba lleno) (3 puntos)
  - Llama, muestra, vuelve a llamar y cuelga el primer móvil del array.
2. En un proyecto TUF7\_Ej2\_tu\_nombre implementa las clases Equipo y Jugador (partiendo de los modelos UML que están anexos abajo). Una vez creadas las clases, implementa un método main que introduzca por teclado los valores de los atributos para crear un equipo. Se preguntará cuántos jugadores tiene un equipo. Una vez introducidos los valores mostrar la información completa del equipo. Los métodos mostrar de cada clase sacarán por pantalla toda la información de los objetos. (4 puntos)

\* La creación de las clases tiene que ser totalmente fiel al modelado UML. Tanto en los métodos creados, nombres de la clase y atributos e incluso su forma de escribirlos (mayúsculas y minúsculas). Cada incumplimiento de este punto restará 0.5 puntos.

<b>Movil</b>
-modelo (String) -precio (float) -hablando (boolean)
+Movil(String, float) +setModelo(String):void +getModelo():String +setPrecio(float):void +llamar():void +colgar():void +mostrarMovil():void

<b>Equipo</b>
-nombre(String) -ciudad(String) -jugadores (Jugador [])
+Equipo() +setNombre(String):void +getNombre():String +setCiudad(String):void +getCiudad():String +setJugadores(Jugador[]):void +getJugadores():Jugador[] +mostrarEquipo():void

<b>Jugador</b>
-nombre(String) -posicion(String) -dorsal (int)
+Jugador() +setNombre(String):void +getNombre():String +setPosicion(String):void +getPosicion():String +setDorsal(int):void +getDorsal():int +mostrarJugador():void