

EJERCICIOS UD 5 COLECCIONES PARTE 1

NOTA: Aunque no se pida explícitamente en cada ejercicio, todos deben tener un main en el que se prueben las funcionalidades realizadas en los métodos, haciendo las llamadas y mostrando los cálculos que se hagan. Se pueden crear y agregar a la colección algunos objetos de prueba antes, para no tener que probar empezando siempre con la lista vacía.

- 1) Realizar un programa que permita gestionar notas de texto (como las de un móvil, no son calificaciones de clase). Usaremos la clase ArrayList y el listado debe guardar objetos tipo Nota (tú decides los atributos que debe tener una nota). Se podrán mostrar de manera individual (sin necesidad de buscar, solo diciendo cuál de la lista mostrada), dar información del número de notas guardadas, agregar una nueva nota, un método que borre una de las notas guardadas comprobando antes que la lista no está vacía. Se borrará por número (la nota 1, la 2...), por lo que habrá que mostrarle al usuario la lista con todas las notas (en un método) y su número correspondiente, el usuario introducirá el número de la que quiera borrar (esto no es lo ideal puesto que así habría que mostrar todas las notas ¿y si hay miles de ellas?).
- 2) Crear un programa para gestionar los socios de un Club de campo. Se debe poder guardar Socios con sus datos personales, buscarlos, mostrar sus datos, modificarlos y borrarlos (operaciones CRUD). Usar la clase ArrayList. Este ejercicio es para empezar, así que no es necesario hacer nada más, aunque se podrían agregar funcionalidades como alquilar pistas, pagar cuotas, etc.

Debemos usar la clase Socio, la clase Club y la principal como mínimo. Puedes probar a separar "las operaciones CRUD" en una clase CRUDSocio de la clase Club.

- 3) Realizar un programa que guarde Alumnos de una clase. Se deben poder mostrar, modificar y borrar los que se quiera de forma individual, es decir, el usuario debe decir quién quiere mostrar, modificar o borrar. Se puede crear otra clase con funcionalidades como sacar nota media del curso, dar el número de suspensos en total, media de suspensos, etc. Usa la interfaz Set implementada por HashSet.
- 4) Realizar, usando la clase HashMap, una agenda de teléfonos (un Contacto como clave y el número de teléfono como valor). Crear aquellos métodos que se utilizan normalmente en una agenda, como agregar, borrar, mostrar, buscar por algún criterio (por ejemplo, por nombre), modificar algún dato, etc. Y un main para probar todo.
- 5) Crear una pequeña lista con los datos personales de algunas Personas, y mostrarla por pantalla de forma NO ordenada y de forma ordenada (elegid la forma de ordenar vosotros, podéis empezar con orden natural, es decir, alfabéticamente y después crear otra "no natural").
- 6) Crear una clase Trabajador con los siguientes atributos: nombre, DNI, Horas trabajadas y sueldo final (este se puede calcular de cualquier manera sencilla de las que hemos visto anteriormente en otros ejercicios).

Guardar varios trabajadores en una colección y mostrar los trabajadores ordenados según el número de horas trabajadas (de menor a mayor y viceversa) y según el sueldo (del que más al que menos cobra). Se deben usar las interfaces comparable y/o comparator.

- 7) Puedes hacer muchos de los ejercicios del tema anterior que usaban arrays, pero usando colecciones en lugar de arrays de objetos.