Lab01

Laboratorio Repaso

**Práctica 01:**

Desarrollar una clase llamada **Password** que siga las siguientes condiciones:

* Que tenga los atributos longitud y contraseña. Por defecto, la longitud sera de 8.
* Los constructores serán los siguiente:
* Un constructor por defecto.
* **Un constructor con la longitud que nosotros le pasemos. Generará una contraseña aleatoria con esa longitud.**
* **Los métodos que implementa serán:**
* **esFuerte**(): devuelve un booleano si es fuerte o no, para que sea fuerte debe tener mas de 2 mayúsculas, mas de 1 minúscula y mas de 5 números.
* **generarPassword**():  genera la contraseña del objeto con la longitud que tenga.
* Método get para contraseña y longitud.
* Método set para longitud.

Ahora, crea una clase ejecutable llamada **Main**:

* Crea un array de Passwords con el tamaño que tu le indiques por teclado, junto con la longitud de los passwords (todos tendrán la misma).
* Crea otro array de booleanos donde se almacene si el password del array de Password es o no

Al final, muestra la contraseña y si es o no fuerte (usa el bucle anterior). Usa este simple formato:

* contraseña1 valor\_booleano1
* contraseña2 valor\_bololeano2

**Práctica 2:**

Vamos a hacer una baraja de cartas españolas orientado a objetos.

Una carta tiene un número entre 1 y 12 (el 8 y el 9 no los incluimos) y un palo (espadas, bastos, oros y copas)

La baraja estará compuesta por un conjunto de cartas, 40 exactamente.

Las operaciones que podrá realizar la baraja son:

barajar: cambia de posición todas las cartas aleatoriamente

**siguienteCarta**: devuelve la siguiente carta que está en la baraja, cuando no haya más o se haya llegado al final, se indica al usuario que no hay más cartas.

**cartasDisponibles**: indica el número de cartas que aún puede repartir

**darCartas**: dado un número de cartas que nos pidan, le devolveremos ese número de cartas (piensa que puedes devolver). En caso de que haya menos cartas que las pedidas, no devolveremos nada pero debemos indicárselo al usuario.

**cartasMonton**: mostramos aquellas cartas que ya han salido, si no ha salido ninguna indicárselo al usuario

**mostrarBaraja**: muestra todas las cartas hasta el final. Es decir, si se saca una carta y luego se llama al método, este no mostrara esa primera carta.