

## En esta unidad has aprendido cómo funcionan las distintas estructuras de almacenamiento (array, ArrayList ,colecciones, etc.)

La tarea consiste en realizar una serie de ejercicios relacionados con los contenidos vistos.

---

### Arrays:

#### Temperatura:

1. Haz Programa que lea por teclado las temperaturas de los 10 últimos días y los almacene en un array de números decimales.

A continuación, muestra e implementa el siguiente menú:

- |                                |
|--------------------------------|
| 1) Temperatura media           |
| 2) Temperatura máxima          |
| 3) Temperatura mínima          |
| 4) Días por encima de la media |
| 5) Salir                       |

- **Temperatura media:** muestra la temperatura media de los últimos 10 días
- **Temperatura máxima:** muestra la máxima temperatura alcanzada.
- **Temperatura mínima:** muestra la mínima temperatura alcanzada.
- **Días por encima de la media:** muestra los días que tuvieron una temperatura por encima de la temperatura media.

#### Lotería primitiva:

2. Disponemos de un boleto de lotería primitiva de 8 apuestas. Guarda en un array 6x8 las 8 combinaciones jugadas. Puedes generar cada combinación con números aleatorios. Por ejemplo, el **array boleto** podría ser:

1	5	14	23	32	45
2	4	6	34	35	36
3	4	5	32	13	23
4	3	34	45	46	47
5	6	23	31	33	36
6	23	12	8	45	32
23	34	12	13	44	33
5	45	49	23	13	43

En otro **array combinacionGanadora** (de 6 casillas) genera una combinación ganadora aleatoria:

12	21	33	25	32	48
----	----	----	----	----	----

Ten en cuenta que:

- Todos los números de la combinación deben ser diferentes.
- Los números estarán comprendidos entre el 1 y el 49.
- `int valor = (int) Math.floor(Math.random()*49+1);` genera un numero aleatorio comprendido desde el 1 al 49. Más información en: <https://www.discoduroderoer.es/como-generar-numeros-aleatorios-con-el-metodo-math-random-de-java/>

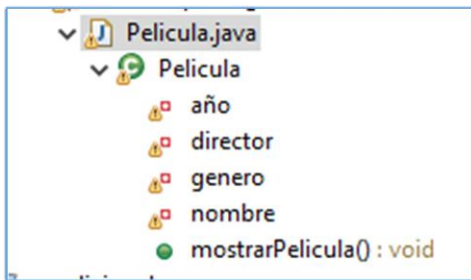
**Simula el juego de la lotería primitiva y calcula el número de aciertos en cada una de las combinaciones del boleto.**

---

## ArrayList:

### Cine:

3. Crea dentro del proyecto Cine, una clase Película con los siguientes atributos y métodos.



(añade los métodos getter, setter y constructores).

A continuación, crea un programa principal con un ArrayList de películas que realice las siguientes acciones:

- 6) Añadir película
- 7) Borrar película
- 8) Mostrar películas
- 9) Buscar por genero
- 10) Buscar por letra
- 11) Salir

**Añadir película:** Añade una nueva película en el ArrayList de películas

**Borrar película:** Borra del ArrayList de películas, la película con el nombre indicado

**Mostrar películas:** Muestra información de todas las películas

**Buscar por género:** Muestra la información de las películas del género dado.

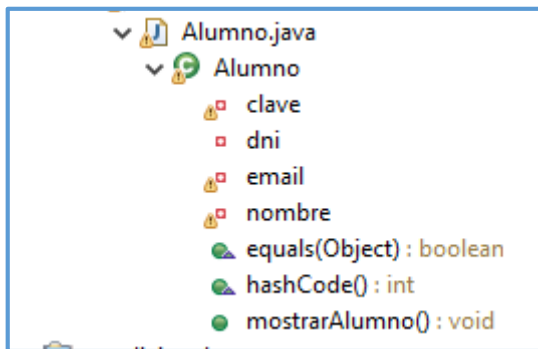
**Buscar por letra:** Muestra la información de las películas que comiencen por la letra especificada.

---

## Colecciones:

**Matriculas:** (utiliza la colección LinkedList)

4. Crea dentro del proyecto Matriculas, una clase Alumno con los siguientes atributos y métodos.



- Añade los métodos getter, setter y el constructor/es.
- Genera los métodos equals y hashCode con el dni.

A continuación, crea un programa principal con dos listas <LinkedList>:

- **AlumnosPreinscritos** -> Una lista con los alumnos que hicieron una preinscripción
- **AlumnosMatriculados** -> Una lista con los alumnos ya matriculados en un curso

- 1) **Matricular alumno**
- 2) **Borrar alumno matriculado con dni**
- 3) **Mostrar ultimo alumno matriculado**
- 4) **Mostrar alumnos matriculados**
- 5) **Salir**

**Matricular Alumno:** Quita de la lista de preinscritos, al alumno que se encuentra en la primera posición y lo inserta al final de la lista de alumnos matriculados.

**Borrar alumno matriculado con dni:** Quita de la lista de alumnos matriculados al alumno con el dni dado. A continuación, matricula al alumno que se encuentre en la primera posición de la lista de preinscripciones.

**Mostrar ultimo alumno matriculado:** Muestra el último alumno matriculado.

**Mostrar alumnos matriculados:** Muestra la lista de alumnos matriculados.

Los ejercicios tienen la siguiente puntuación:

- **Ejercicio 1 -> 2,5 puntos**
- **Ejercicio 2 -> 2,5 puntos**
- **Ejercicio 3 -> 2,5 puntos**
- **Ejercicio 4 -> 2,5 puntos**

Entrega un fichero comprimido con todo el código del proyecto y dale un nombre similar a éste: `sanchez_manas_begona_tarea8.1` (si la alumna se llama Begoña Sanchez Mañas)

