

## Trabajo Práctico – Array de Objetos

Desarrolle los programas de manera tal que se permita cargar una cantidad fija de objetos o que esa cantidad se solicite por teclado. Recuerde que las clases propuestas, deben escribirse en su propio archivo .java.

1. Crear un programa Java que permita cargar un array de objetos de clase Automovil. Una vez cargado el array, se deberá informar la patente, propietario, modelo y kilometraje del automóvil de mayor kilometraje.

```
public class Automovil {  
    private String patente;  
    private String propietario;  
    private int modelo;  
    private long kilometraje;  
    //defina los métodos  
}
```

2. Desarrollar un programa que permita generar un array de objetos de clase Computadora. Luego de cargado el array se deberá procesarlo para determinar la cantidad de Computadoras de cada tipo.

```
public class Computadora {  
    private int numero; // id computadora  
    private int anio; // año de adquisición  
    private int tipo; // tipo de computadora (valores válidos:1 a 5)  
    //defina los métodos  
}
```

3. Crear un programa que permita cargar un array de objetos de clase Docente y luego determinar un promedio de los sueldos básicos cargados y los datos del docente con menor sueldo básico.

```
public class Docente {  
    private long dni;  
    private String apelnomb;  
    private double sueldoBasico;  
    //defina los métodos  
}
```

4. Desarrollar un programa que permita generar un array de objetos de clase Prestador, la cual corresponde a un Prestador Médico. Luego de cargado el array se deberá procesarlo para determinar qué especialidad es la que tiene más Prestadores Médicos.

```
public class Prestador {  
    private int matricula;  
    private int especialidad; // especialidad de prestador (valores: 1 a 4)  
    //defina los métodos  
}
```

5. Un instituto de computación cuenta con un gabinete con 10 computadoras. Se requiere un programa para automatizar la reserva de los equipos para prácticas. Para ello contará con un array de objetos de clase PC de 10 elementos en el cual se almacenará el estado de cada equipo: 0 = no reservado , 1 = reservado. El programa deberá presentar un menú opciones que permitirá reservar o cancelar una reserva y visualizar el estado de las 10 computadoras. El menú de opciones será:

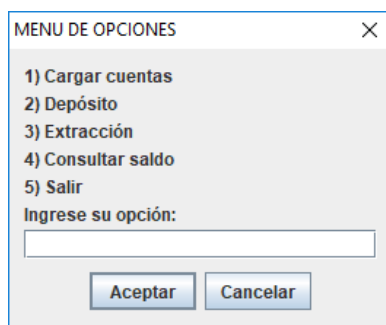
1. Asignar o cancelar reserva.
2. Visualizar estado de las computadoras.
3. Salir

1. Opción Asignar o cancelar reserva: deberá solicitar el nro de computadora y si no está reservada, preguntará si desea reservarla. En caso de que esté reservada, preguntará si desea cancelar la reserva.
2. Opción Visualizar estado de las computadoras: mostrará una lista de las computadoras, indicando número de computadora y su estado (reservada, no reservada).

```
public class PC {  
    private int numero ;  
    private int estado;  
    //defina los métodos  
}
```

**Tenga en cuenta que, para operar con una computadora, deberá ingresar el número de la misma y verificar su existencia. Se aconseja crear un método que reciba como parámetro un número de computadora para buscarlo en el array y retorne un valor que indique o no su existencia.**

6. Desarrollar un programa para simular depósitos y/o extracciones de un grupo de Cajas de Ahorro. Para esto se deberá generar un menú de opciones mediante `JOptionPane.showInputDialog`.. Este menú deberá ser similar a:



- La opción 1 deberá solicitar los datos de las cuentas. Las cuentas se van almacenar en una array de clase Cuenta. La cantidad de cuentas podrá ser un valor fijo definido en el programa o un valor ingresado por teclado.
- La opción 2 permitirá realizar un Depósito en una cuenta.
- La opción 3 permitirá realizar una Extracción de una cuenta. Informar si la cuenta no posee saldo para esta operación.
- La opción 4 mostrará el saldo de una cuenta.

```
public class Cuenta {  
    private int numero;  
    private String titular;  
    private double saldo ;  
    //defina los métodos  
}
```

**Tenga en cuenta que, para operar con una cuenta, deberá ingresar el número de la misma y verificar su existencia. Se aconseja crear un método que reciba como parámetro un número de cuenta para buscarlo en el array y retorne un valor que indique o no su existencia.**