

[illegible]

```

        listaPersonas.add(profe);
        break;
    case 3:
        empleado.setTipo(3);
        input.nextLine();
        System.out.print("Nombre : ");
        empleado.setNombre(input.nextLine());
        System.out.print("Apellido : ");
        empleado.setApellido(input.nextLine());
        System.out.print("DNI : ");
        empleado.setDni(input.nextLine());
        System.out.print("Estado civil : ");
        empleado.setEstadoCivil(input.nextLine());
        System.out.print("Año incorporacion : ");
        empleado.setAgeIncorporacion(input.nextInt());
        System.out.print("Despacho : ");
        empleado.setNumDespacho(input.nextInt());
        listaPersonas.add(empleado);
        break;
    case 4:
        for (Persona per : listaPersonas) {
            System.out.println("Nombre: " + per.getNombre() + " Apellido: " + per.apellido + " DNI: " +
per.getDni() + " Estado Civil: " + per.getEstadoCivil() + " Año incorporacion: " + per.getAgeIncorporacion());
        }
        break;
    case 5:
        for (Persona per : listaPersonas) {
            do {
                System.out.print("1: sig 2: cancel opcion : ");
                opcion = input.nextInt();
                System.out.println("Nombre: " + per.getNombre() + " Apellido: " + per.apellido + " DNI: " +
per.getDni() + " Estado Civil: " + per.getEstadoCivil() + " Año incorporacion: " + per.getAgeIncorporacion());
                break;
            } while (opcion != 2);
        }
        break;
    case 6:
        for (int i = 0; i < listaPersonas.size(); i++) {
            listaPersonas.get(i).getTipo();
            System.out.print("Eliminar --> 1: Alumno 2: Profesor 3: Empleado opcion : ");
            opcion = input.nextInt();
            switch (opcion) {
                case 1:
                    if (listaPersonas.get(i).getTipo() == 1);
                    listaPersonas.remove(i);
                    break;
                case 2:
                    if (listaPersonas.get(i).getTipo() == 2);
                    listaPersonas.remove(i);
                    break;
                case 3:
                    if (listaPersonas.get(i).getTipo() == 3);
                    listaPersonas.remove(i);
                    break;
            }
        }
        break;
    case 7 :break;

```

```

        default:
            System.out.println("Elija numero del 1 al 6");
        }
    }
    catch(InputMismatchException e){
        System.out.println("dato incorrecto");
        break;
    }
}while (opcion != 7 );
}
}

```

```

package ej1;

```

```

public class Alumno extends Persona{

    public Alumno(){

    }
}

```

```

package ej1;

```

```

public abstract class Persona {
    protected String nombre , apellido , dni , estadoCivil ;
    private int ageIncorporacion , tipo;

    public Persona(){

    }

    public String getNombre() {
        return nombre;
    }

    public void setNombre(String nombre) {
        this.nombre = nombre;
    }

    public String getApellido() {
        return apellido;
    }

    public void setApellido(String apellido) {
        this.apellido = apellido;
    }

    public String getDni() {
        return dni;
    }

    public void setDni(String dni) {
        this.dni = dni;
    }

    public String getEstadoCivil() {
        return estadoCivil;
    }
}

```

```

public void setEstadoCivil(String estadoCivil) {
    this.estadoCivil = estadoCivil;
}

public int getAgeIncorporacion() {
    return ageIncorporacion;
}

public void setAgeIncorporacion(int ageIncorporacion) {
    this.ageIncorporacion = ageIncorporacion;
}

public int getTipo() {
    return tipo;
}

public void setTipo(int tipo) {
    this.tipo = tipo;
}
}

```

```

package ej1;
public class Profesor extends Persona{
    private int numOficina;

    public Profesor(){

    }

    public int getNumOficina() {
        return numOficina;
    }
    public void setNumOficina(int numOficina) {
        this.numOficina = numOficina;
    }
}

```

```

package ej1;

public class Empleado extends Persona{
    private int numDespacho;

    public Empleado(){

    }

    public int getNumDespacho() {
        return numDespacho;
    }
    public void setNumDespacho(int numDespacho) {
        this.numDespacho = numDespacho;
    }
}

```

Ejercicio 2 - ArrayList Programa que permite insertar N datos de Productos en un ArrayList (los datos de cada Producto son Nombre, código, stock). Posteriormente visualice los datos de los productos cuyo stock sea menor a 10.

```
package ej2;

import java.util.ArrayList;
import java.util.InputMismatchException;
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        int opcion = 0;
        int stockInput = 0;
        boolean continuar = false;
        Scanner input = new Scanner(System.in);
        ArrayList<Productos> listaProductos = new ArrayList<Productos>();
        do {
            System.out.print("1: Registrar productos  2: Ver Productos  3: Salir  opcion--> ");
            opcion = input.nextInt();
            switch (opcion) {
                case 1:
                    Productos producto = new Productos();
                    input.nextLine();
                    System.out.print("Nombre producto : ");
                    producto.setNombre(input.nextLine());
                    System.out.print("Codigo : ");
                    producto.setCodigo(input.nextLine());
                    do {
                        Scanner inputBucle2 = new Scanner(System.in);
                        System.out.print("Stock : ");
                        try {
                            stockInput = inputBucle2.nextInt();
                            producto.setStock(stockInput);
                            listaProductos.add(producto);
                            continuar = true;
                        } catch (InputMismatchException ignored) {
                            System.out.println("Debe ingresar un numero");
                        }
                    } while (!continuar);
                    break;
                case 2:
                    for (Productos p : listaProductos) {
                        if (p.getStock() < 10) {
                            System.out.println("Nombre: " + p.getNombre() + "  Codigo: " + p.getCodigo() + "  Stock: " +
p.getStock());
                            System.out.println("_____");
                        }
                    }
                    break;
            }
        } while (opcion != 3);
    }
}
```

```
package ej2;
```

```

public class Productos {
    private String nombre , codigo;
    private int stock;

    public Productos(){

    }

    public String getNombre() {
        return nombre;
    }

    public void setNombre(String nombre) {
        this.nombre = nombre;
    }

    public String getCodigo() {
        return codigo;
    }

    public void setCodigo(String codigo) {
        this.codigo = codigo;
    }

    public int getStock() {
        return stock;
    }

    public void setStock(int stock) {
        this.stock = stock;
    }
}

```

Ejercicio 3 – Arreglos estáticos Realizar el juego busca minas, el cual debe tener un arreglo multidimensional estático de 10 x 10, deberán buscar e investigar la clase Random, para poder asignar en donde van ha estar posicionadas las minas dentro de la matriz. Luego permitir al usuario que pueda ir seleccionando algún espacio del arreglo hasta que encuentre una mina o descubra todos los lugares.

```

package ej3;
import java.util.InputMismatchException;
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner leer = new Scanner(System.in);

        int tablero[][] = new int[10][10];

        double w = 0 , z = 0;
        int ntminas = 0 , intento = 30 , tminas = 10;
        boolean continuar = false;

        for (int j = 0; j < 10; j++)
            for (int i = 0; i < 10; i++)

```

```

        tablero[j][i] = 0;
    }
} while (ntminas <= 5);

System.out.println("\n-x-----x-");
System.out.println(" Ingrese las coordenadas en el rango del 1 al 10 ");
do {
    int x = 0, y = 0, nminas = 0, cor;
    do {
        do {
            Scanner cordenada = new Scanner(System.in);
            try {
                System.out.println("\n-x-----x-");
                System.out.println("\n          Intento : " + intento + " , Cual es su coordenada? \n");
                System.out.print("          Fila: ");
                cor = cordenada.nextInt();
                x = cor;
                System.out.println();
                System.out.print("          Columna: ");
                cor = cordenada.nextInt();
                y = cor;
                continuar = true;
            }
            catch (InputMismatchException ex){
                System.out.println("Ingrese un numero");
            }
        } while(!continuar);
        if (x == 0 || y == 0 || x > 10 || y > 10)
            System.out.println("\n ==>Coordenadas incorrectas<== \n");
    } while (x == 0 || y == 0 || x > 10 || y > 10);

    x = x - 1;
    y = y - 1;

    if (tablero[x][y] == 0) {
        if (x == 0 && y == 0) {
            if (x + 1 >= 0 && x + 1 < 10 && y < 10)
                nminas = nminas + tablero[x + 1][y];
            if (x + 1 >= 0 && x + 1 < 10 && y + 1 < 10 && y + 1 >= 0)
                nminas = nminas + tablero[x + 1][y + 1];
            if (x >= 0 && x < 10 && y + 1 < 10 && y + 1 >= 0)
                nminas = nminas + tablero[x][y + 1];
        } else {
            if (y == 0) {
                if (x + 1 >= 0 && x + 1 < 10 && y < 10)
                    nminas = nminas + tablero[x + 1][y];
                if (x - 1 >= 0 && x - 1 < 10 && y < 10)
                    nminas = nminas + tablero[x - 1][y];
                if (x - 1 >= 0 && x - 1 < 10 && y + 1 < 10 && y + 1 >= 0)
                    nminas = nminas + tablero[x - 1][y + 1];
            }
        }
    }
}

```

```

        if (x >= 0 && x < 10 && y + 1 < 10 && y + 1 >= 0)
            nminas = nminas + tablero[x][y + 1];
        if (x + 1 >= 0 && x + 1 < 10 && y + 1 < 10 && y + 1 >= 0)
            nminas = nminas + tablero[x + 1][y + 1];
    } else {
        if (x - 1 >= 0 && x - 1 < 10 && y - 1 < 10)
            nminas = nminas + tablero[x - 1][y - 1];
        if (x >= 0 && x < 10 && y - 1 < 10)
            nminas = nminas + tablero[x][y - 1];
        if (x + 1 >= 0 && x + 1 < 10 && y - 1 < 10)
            nminas = nminas + tablero[x + 1][y - 1];
        if (x + 1 >= 0 && x + 1 < 10 && y < 10)
            nminas = nminas + tablero[x + 1][y];
        if (x - 1 >= 0 && x - 1 < 10 && y < 10)
            nminas = nminas + tablero[x - 1][y];
        if (x - 1 >= 0 && x - 1 < 10 && y + 1 < 10 && y + 1 >= 0)
            nminas = nminas + tablero[x - 1][y + 1];
        if (x >= 0 && x < 10 && y + 1 < 10 && y + 1 >= 0)
            nminas = nminas + tablero[x][y + 1];
        if (x + 1 >= 0 && x + 1 < 10 && y + 1 < 10 && y + 1 >= 0)
            nminas = nminas + tablero[x + 1][y + 1];
    }
}
System.out.print("\n      Mina no encontrada ");

if (intento > tminas)
    System.out.print("      Alrededor hay: " + nminas);
System.out.println("\n      Intentos restantes: " + (intento - 1));
} else {
    tminas = tminas - 1;
    System.out.println("\n      Acertaste!, Mina Eliminada quedan " + tminas + " Minas");
    System.out.println("\n      Intentos restantes: " + (intento - 1));
}
intento = intento - 1;
} while (intento >= tminas && intento > 0 && tminas > 0);

if (tminas == 0)
    System.out.println("      GANASTE!");

if (tminas > intento)
    System.out.println("      PERDISTE! (Exceso de minas restantes " + tminas + ")\n\n");

if (intento == 0)
    System.out.println("      PERDISTE!");
}
}

```