

```

////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
package exame;

public class CPaciente {
    private String codigo;
    private String nombre;
    private int habitacion;
    private float temperatura;

    public CPaciente(String cod, String nom, int hab, float tem) {
        codigo = cod;
        nombre = nom;
        habitacion = hab;
        temperatura = tem;
    }

    public void setCodigo(String cod) {
        codigo = cod;
    }

    public String getCodigo() {
        return codigo;
    }

    public void setNombre(String nom) {
        nombre = nom;
    }

    public String getNombre() {
        return nombre;
    }

    public void setHabitacion(int hab) {
        habitacion = hab;
    }

    public int getHabitacion() {
        return habitacion;
    }

    public void setTemperatura(float tem) {
        temperatura = tem;
    }

    public float getTemperatura() {
        return temperatura;
    }

    public void mostrar() {
        System.out.print(codigo + "\t" + nombre + "\t\t");
        System.out.println(habitacion + "\t\t" + temperatura);
    }
}

```

```

////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
package exame;

import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.io.EOFException;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;

public class CHospital {
    private CPaciente[] hospital;
    private int total;
    private DataOutputStream flujoSalida;
    private DataInputStream flujoEntrada;
    private File fichero;

    public CHospital(File fich) {
        hospital = new CPaciente[10];
        total = 0;
        fichero = fich;
    }

    public void llenar() {
        hospital[0] = new CPaciente("P01", "Ignacio Arias", 25, 38.2F);
        hospital[1] = new CPaciente("P02", "Ana Menéndez", 39, 39.1F);
        hospital[2] = new CPaciente("P03", "Lucas García", 32, 38.6F);
        hospital[3] = new CPaciente("P04", "Mercedes Novoa", 18, 37.9F);
        hospital[4] = new CPaciente("P05", "María Argüelles", 15, 38.7F);
        total = 5;
    }

    public void vaciar() {
        total = 0;
    }

    public void salvar() {
        int i;

        if (total == 0)
            System.out.println("Hospital VACIO");
        else {
            flujoSalida = null;
            try {
                flujoSalida = new DataOutputStream(
                    new FileOutputStream(fichero));
                for (i = 0; i < total; i++) {
                    flujoSalida.writeUTF(hospital[i].getCodigo());
                    flujoSalida.writeUTF(hospital[i].getNombre());
                    flujoSalida.writeInt(hospital[i].getHabitacion());
                    flujoSalida.writeFloat(hospital[i].getTemperatura());
                }
            } catch (IOException e) {
                System.out.println("problemas al SALVAR el fichero");
            }
        }
    }
}

```

```

        } finally {
            if (flujoSalida != null)
                try {
                    flujoSalida.close();
                } catch (IOException e) {
                    System.out.println(
                        "problemas al CERRAR el flujo de salida");
                }
        }
    }
}

public void restaurar() {
    String codigo;
    String nombre;
    int habitacion;
    float temperatura;
    int i;

    flujoEntrada = null;
    i = 0;
    try {
        flujoEntrada = new DataInputStream(
            new FileInputStream(fichero));
        do {
            codigo = flujoEntrada.readUTF();
            nombre = flujoEntrada.readUTF();
            habitacion = flujoEntrada.readInt();
            temperatura = flujoEntrada.readFloat();
            hospital[i] = new CPaciente(codigo, nombre,
                                       habitacion, temperatura);
            i++;
        } while (true);
    } catch (EOFException e) {
        // no se hace nada
    } catch (IOException e) {
        System.out.println("problemas al RESTAURAR el fichero");
    } finally {
        if (flujoEntrada != null)
            try {
                flujoEntrada.close();
            } catch (IOException e) {
                System.out.println("problemas al CERRAR
                                   el flujo de entrada");
            }
    }
    total = i;
}

private CPaciente damePaciente(String cod) {
    int i;

    i = 0;
    while (i < total)
        if (hospital[i].getCodigo().equals(cod))
            return hospital[i];
        else
            i++;
    return null;
}

```

```

private int dameIndice(String cod) {
    int i;

    i = 0;
    while (i < total)
        if (hospital[i].getCodigo().equals(cod))
            return i;
        else
            i++;
    return -1;
}
private void elimina(int i) {
    while (i < total - 1) {
        hospital[i] = hospital[i + 1];
        i++;
    }
    total--;
}
public void eliminarPaciente(String codigo) {
    int indice;

    indice = dameIndice(codigo);
    if (indice == -1)
        System.out.println("Paciente no ENCONTRADO");
    else {
        elimina(indice);
        System.out.println("Paciente ELIMINADO");
    }
}

public void insertarPaciente() {
    String codigo;
    String nombre;
    int habitacion;
    float temperatura;

    System.out.print("Dame código: ");
    codigo = Leer.datoString();
    System.out.print("Dame nombre: ");
    nombre = Leer.datoString();
    System.out.print("Dame habitacion: ");
    habitacion = Leer.datoInt();
    System.out.print("Dame temperatura: ");
    temperatura = Leer.datoFloat();

    hospital[total] = new CPaciente(codigo, nombre,
                                    habitacion, temperatura);
    total++;
}
public void modificarPaciente(String codigo) {
    CPaciente paciente;
    float temperatura;

    paciente = damePaciente(codigo);
    if (paciente == null)
        System.out.println("Paciente no ENCONTRADO");
    else {
        System.out.println("Temperatura ACTUAL: " +
                           paciente.getTemperatura());
        System.out.print("Dame temperatura NUEVA: ");
        temperatura = Leer.datoFloat();
    }
}

```

```

        paciente.setTemperatura(temperatura);
        System.out.println("Paciente MODIFICADO");
    }
}

public void mostrarPaciente(String codigo) {
    CPaciente paciente;

    paciente = damePaciente(codigo);
    if (paciente == null)
        System.out.println("Paciente no ENCONTRADO");
    else {
        mostrarCabecera();
        paciente.mostrar();
    }
}

public void mostrar() {
    int i;

    if (total == 0)
        System.out.println("Hospital VACIO");
    else {
        mostrarCabecera();
        for (i = 0; i < total; i++)
            hospital[i].mostrar();
    }
}

private void mostrarCabecera() {
    System.out.println("CODIGO" + "\tNOMBRE" +
        "\t\tHABITACION" + "\tTEMPERATURA");
    System.out.println("=====" + "\t=====" +
        "\t\t=====" + "\t=====");
}

private void menu() {
    System.out.println("\n M E N U - Exame");
    System.out.println(" =====");
    System.out.println("1.- LLENAR Hospital");
    System.out.println("2.- VACIAR Hospital");
    System.out.println("3.- MOSTRAR Hospital");
    System.out.println("4.- MOSTRAR Paciente");
    System.out.println("5.- INSERTAR Paciente");
    System.out.println("6.- MODIFICAR Paciente");
    System.out.println("7.- ELIMINAR Paciente");
    System.out.println("8.- SALVAR Hospital en Fichero");
    System.out.println("9.- RESTAURAR Hospital desde Fichero");
    System.out.println("10.- FINAL");
    System.out.print("Pulsa opción: ");
}

public int dameOpcion() {
    int opcion;

    menu();
    opcion = Leer.datoInt();
    System.out.println();
    return opcion;
}
}

```

```

////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
package exame;

import java.io.File;

public class proba {

    public static void main(String[] args) {
        CHospital hospital;
        File fichero;
        int numero;
        String codigo;

        fichero=new File("exame.dat");
        hospital = new CHospital(fichero);
        do {
            numero = hospital.dameOpcion();
            switch (numero) {
                case 1:
                    hospital.llenar();
                    System.out.println("Hospital LLENO");
                    break;
                case 2:
                    hospital.vaciar();
                    System.out.println("Hospital VACIO");
                    break;
                case 3:
                    hospital.mostrar();
                    break;
                case 4:
                    System.out.print("Dame código de paciente: ");
                    codigo = Leer.datoString();
                    hospital.mostrarPaciente(codigo);
                    break;
                case 5:
                    hospital.insertarPaciente();
                    break;
                case 6:
                    System.out.print("Dame código de paciente: ");
                    codigo = Leer.datoString();
                    hospital.modificarPaciente(codigo);
                    break;
                case 7:
                    System.out.print("Dame código de paciente: ");
                    codigo = Leer.datoString();
                    hospital.eliminarPaciente(codigo);
                    break;
                case 8:
                    hospital.salvar();
                    System.out.println("Hospital SALVADO en fichero");
                    break;
                case 9:
                    hospital.restaurar();
                    System.out.println("Hospital RESTAURADO desde fichero");
                    break;
                case 10:
                    System.out.println("FINAL");
                    break;
            }
        }
    }
}

```

```

        default:
            System.out.println("OPCION Erronea");
            break;
        }
    } while (numero != 10);
}

////////////////////////////////////
////////////////////////////////////
package exame;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

public class Leer {

    public static String dato() {
        String sdato = "";
        try {
            // Definir un flujo de caracteres de entrada: flujoE
            InputStreamReader isr = new InputStreamReader(System.in);
            BufferedReader flujoE = new BufferedReader(isr);
            // Leer. La entrada finaliza al pulsar la tecla Entrar
            sdato = flujoE.readLine();
        } catch (IOException e) {
            System.err.println("Error: " + e.getMessage());
        }
        return sdato; // devolver el dato tecleado
    }

    public static char datoChar() {
        return dato().charAt(0);
    }

    public static String datoString() {
        return dato();
    }

    public static short datoShort() {
        try {
            return Short.parseShort(dato());
        } catch (NumberFormatException e) {
            System.out.print("Dato short INVALIDO. Teclee otro: ");
            return datoShort();
        }
    }

    public static int datoInt() {
        try {
            return Integer.parseInt(dato());
        } catch (NumberFormatException e) {
            System.out.print("Dato int INVALIDO. Teclee otro: ");
            return datoInt();
        }
    }
}

```

```

public static long datoLong() {
    try {
        return Long.parseLong(dato());
    } catch (NumberFormatException e) {
        System.out.print("Dato long INVALIDO. Teclee otro: ");
        return datoLong();
    }
}

public static float datoFloat() {
    try {
        return Float.parseFloat(dato());
    } catch (NumberFormatException e) {
        System.out.print("Dato float INVALIDO. Teclee otro: ");
        return datoFloat();
    }
}

public static double datoDouble() {
    try {
        return Double.parseDouble(dato());
    } catch (NumberFormatException e) {
        System.out.print("Dato double INVALIDO. Teclee otro: ");
        return datoDouble();
    }
}
}

```