

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package Examen.yotelollevo;

import java.io.Serializable;

/**
 *
 * @author anusk
 */
public class CodigoPostal implements Serializable {

    private String cp;

    public CodigoPostal(String cp) throws CPEException {
        if (esValido(cp))
            this.cp = cp;
        else
            throw new CPEException();
    }

    public static boolean esValido(String cp) {
        if (cp.length() != 5)
            return false;
        try {
            int numero = Integer.valueOf(cp);
            if (numero >= 0 && numero <= 99999)
                return true;
            else
                return false;
        } catch (NumberFormatException e) {
            return false;
        }
    }

    public String getCodigoPostal() {
        return cp;
    }

}

```

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
```

```
package Examen.yotelollevo;
```

```
/**
 *
 * @author anusk
 */
public class CPEException extends Exception {

    public CPEException() {
    }
    public CPEException(String m) {
        super(m);
    }

}
```

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package Examen.yotelollevo;

import java.io.Serializable;

/**
 *
 * @author anusk
 */
public abstract class Envio implements Serializable {

    protected int nEnvio;
    protected String nombreR;
    protected String dirR;
    protected CodigoPostal cpR;
    protected String nombreD;
    protected String dirD;
    protected CodigoPostal cpD;

    private static int contador = 1;

    public Envio(String nr, String dr, CodigoPostal cr, String nd, String dd,
        CodigoPostal cd) {
        nEnvio = contador++;
        nombreR = nr;
        dirR = dr;
        cpR = cr;
        nombreD = nd;
        dirD = dd;
        cpD = cd;
    }

    public static int getContador() {
        return contador;
    }

    public static void setContador(int contador) {
        Envio.contador = contador;
    }

    public int getnEnvio() {
        return nEnvio;
    }

    @Override

```

```
public String toString() {  
    return nEnvio + "\tRemitente: " + nombreR + "\t" + dirR + "\t"  
        + cpR.getCodigoPostal() + "\tDestinatario: " + nombreD + "\t"  
        + dirD + "\t" + cpD.getCodigoPostal();  
}  
}
```

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package Examen.yotelollevo;

/**
 *
 * @author anusk
 */
public class EOrdinario extends Envio {

    public EOrdinario(String nr, String dr,CodigoPostal cr, String nd,
        String dd, CodigoPostal cd) {
        super(nr, dr, cr, nd, dd, cd);
    }
}
```

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package Examen.yotelollevo;

import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.IOException;

/**
 *
 * @author anusk
 */
public class ESeguimiento extends Envio implements ISeguimiento {

    public ESeguimiento(String nr, String dr,CodigoPostal cr, String nd,
        String dd, CodigoPostal cd) {
        super(nr, dr, cr, nd, dd, cd);
    }

    @Override
    public String generarCodigo() {
        String ne = String.valueOf(nEnvio);
        while (ne.length() < 5)
            ne = "0" + ne;
        return "PK" + ne + cpD.getCodigoPostal();
    }

    @Override
    public void mostrarSeguimiento() {
        System.out
            .println("Mostrando seguimiento del envío " + generarCodigo());
        BufferedWriter bw = null;
        try {
            bw = new BufferedWriter(new FileWriter(generarCodigo() + ".txt"));
            bw.write("Mostrando seguimiento del envío " + generarCodigo());
        } catch (IOException e) {
            System.out.println("Error escribiendo el fichero de texto");
        } finally {
            try {
                if (bw != null)
                    bw.close();
            } catch (IOException e) {
                System.out.println("Error cerrando el fichero de texto");
            }
        }
    }
}

```

```
}  
  
@Override  
public String toString() {  
    return super.toString() + "\t" + generarCodigo();  
}  
  
}
```

```
/*  
 * To change this license header, choose License Headers in Project Properties.  
 * To change this template file, choose Tools | Templates  
 * and open the template in the editor.  
 */
```

```
package Examen.yotelollevo;
```

```
/**
```

```
 *
```

```
 * @author anusk
```

```
 */
```

```
public interface ISeguimiento {
```

```
    public String generarCodigo();
```

```
    public void mostrarSeguimiento();
```

```
}
```



```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
```

```
package Examen.yotelollevo;
```

```
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.ObjectInputStream;
import java.io.ObjectOutputStream;
import java.util.ArrayList;
import java.util.Iterator;
import java.util.Scanner;
```

```
/**
 *
 * @author anusk
 */
```

```
public class Mensajeria {
```

```
    public static ArrayList<Envio> envios = new ArrayList<Envio>();
    public static final String FDATOS = "envios.bin";
```

```
    public static int leerInt() {
        Scanner sc = new Scanner(System.in);
        while (true) {
            try {
                return sc.nextInt();
            } catch (Exception e) {
                System.out.println("Número no válido");
                sc.nextLine();
            }
        }
    }
}
```

```
    public static int menu() {
        int op = 0;
        while (op < 1 || op > 5) {
            System.out.println("1. Nuevo envío");
            System.out.println("2. Envío entregado");
            System.out.println("3. Listado de envíos");
            System.out.println("4. Mostrar seguimiento");
            System.out.println("5. Salir");
            op = leerInt();
        }
        return op;
    }
}
```

```

    }

    public static void nuevoEnvio() {
        Scanner sc = new Scanner(System.in);
        System.out.print("Nombre del remitente: ");
        String nr = sc.nextLine();
        System.out.print("Dirección del remitente: ");
        String dr = sc.nextLine();
        String cr = "";
        while (!CodigoPostal.esValido(cr)) {
            System.out.print("Código postal del remitente: ");
            cr = sc.nextLine();
        }
        System.out.print("Nombre del destinatario: ");
        String nd = sc.nextLine();
        System.out.print("Dirección del destinatario: ");
        String dd = sc.nextLine();
        String cd = "";
        while (!CodigoPostal.esValido(cd)) {
            System.out.print("Código postal del destinatario: ");
            cd = sc.nextLine();
        }
        int op = 0;
        while (op != 1 && op != 2) {
            System.out.println(
                "¿Qué tipo de envío desea realizar?\n1. Ordinario\n2. Con
seguimiento");
            op = leerInt();
        }
        try {
            if (op == 1) {
                envios.add(new EOrdinario(nr, dr, new CodigoPostal(cr), nd, dr,
                    new CodigoPostal(cd)));
                System.out.println("Se ha añadido el envío");
            } else {
                ESeguimiento e = new ESeguimiento(nr, dr, new CodigoPostal(cr)
                    ,
                    nd, dr, new CodigoPostal(cd));
                envios.add(e);
                System.out.println(
                    "Se ha generado un envío con código de seguimiento "
                    + e.generarCodigo());
            }
        } catch (CPEException e) {
            System.out.println(
                "No se ha podido añadir el envío por un error en el código
postal");
        }
    }
}

```

```

    }

    public static Envio buscar(int n) {
        Iterator<Envio> it = envios.iterator();
        while (it.hasNext()) {
            Envio e = it.next();
            if (e.getnEnvio() == n)
                return e;
        }
        return null;
    }

    public static void envioEntregado() {
        System.out.print("Número de envío: ");
        int n = leerInt();
        Envio e = buscar(n);
        if (e != null) {
            envios.remove(e);
            System.out.println("Se ha entregado el envío " + n);
        } else
            System.out.println("No existe el envío " + n);
    }

    public static void listadoEnvios() {
        Iterator<Envio> it = envios.iterator();
        while (it.hasNext()) {
            System.out.println(it.next());
        }
    }

    public static void mostrarSeguimiento() {
        System.out.print("Número de envío: ");
        int n = leerInt();
        Envio e = buscar(n);
        if (e != null) {
            if (e instanceof ESeguimiento)
                ((ESeguimiento) e).mostrarSeguimiento();
            else
                System.out.println("El envío " + n + " no tiene seguimiento");
        } else
            System.out.println("No existe el envío " + n);
    }

    public static void guardarDatos() {
        ObjectOutputStream oos = null;
        try {
            oos = new ObjectOutputStream(new FileOutputStream(FDATOS));
            oos.writeObject(envios);
        }
    }

```

```

        oos.writeInt(Envio.getContador());
    } catch (IOException e) {
        System.out.println("Error guardando datos");
    } finally {
        try {
            if (oos != null) {
                oos.close();
            }
        } catch (IOException e) {
            System.out.println("Error cerrando el fichero");
        }
    }
}

public static void recuperarDatos() {
    ObjectInputStream ois = null;
    try {
        ois = new ObjectInputStream(new FileInputStream(FDATOS));
        envios = (ArrayList<Envio>) ois.readObject();
        Envio.setContador(ois.readInt());
    } catch (ClassNotFoundException e) {
        System.out.println("Error en los datos");
    } catch (FileNotFoundException e) {
    } catch (IOException e) {
        System.out.println("Error leyendo el fichero");
    } catch (Exception e) {
        System.out.println("Error en los datos");
    } finally {
        try {
            if (ois != null)
                ois.close();
        } catch (IOException e) {
            System.out.println("Error cerrando el fichero");
        }
    }
}

public static void main(String[] args) {

    recuperarDatos();
    int op = menu();
    while (op != 5) {
        switch (op) {
            case 1 :
                nuevoEnvio();
                break;
            case 2 :
                envioEntregado();

```

```
        break;
    case 3 :
        listadoEnvios();
        break;
    case 4 :
        mostrarSeguimiento();
        break;
    }
    op = menu();
}
guardarDatos();

}

}
```