

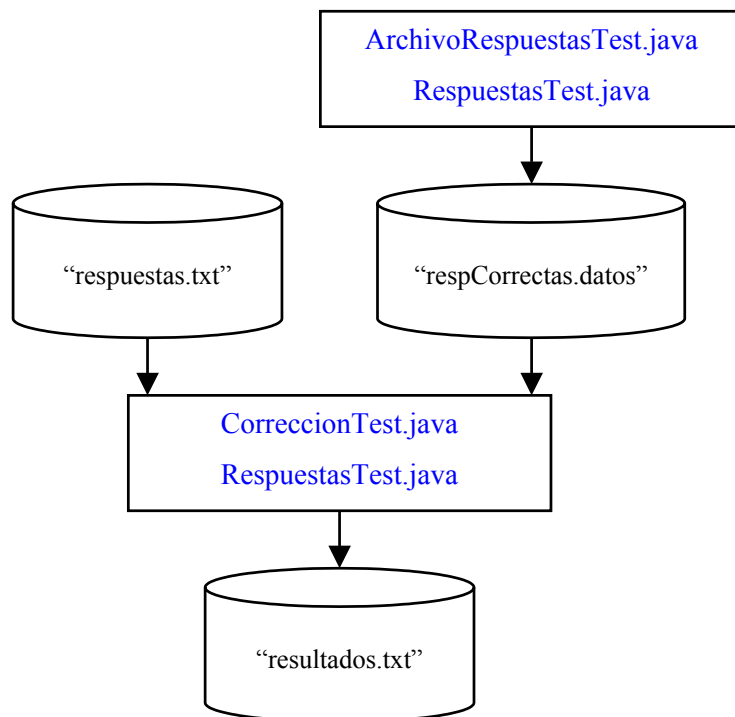
PROGRAMACIÓN I

FECHA: 21-junio-2002

Programa de ficheros de E/S (lau puntu)

Realizar una aplicación en Java para simular la corrección de tests que se llevan a cabo, por ejemplo, para conseguir el carnet de conducir con respecto a la parte teórica. En nuestro caso vamos a suponer que hay cuatro tests diferentes, con la siguiente numeración: 10, 20, 30 y 40.

Esta aplicación se compone de 3 programas o clases, según el siguiente esquema:



Se pide:

1. Estudiar detalladamente los datos reales y los métodos que aparecen en el código fuente de ArchivoRespuestasTest.java. En base a ello, diseñar y codificar cómo sería la clase RespuestasTest, incluyendo como mínimo:
 - los atributos,
 - el/los constructor/es, y
 - los métodos públicosque se necesitan para obtener el fichero de objetos “respCorrectas.datos”.

```
import java.io.*;
```

```

/* Programa ArchivoRespuestasTest.java */

class ArchivoRespuestasTest
{
    public static void main (String[] args) throws IOException
    {
        File ficheroSalida      = new File( "respCorrectas.datos" );
        FileOutputStream flujoSalidaFichero = new FileOutputStream(
            ficheroSalida );
        ObjectOutputStream flujoSalidaObjeto = new ObjectOutputStream(
            flujoSalidaFichero );

        char [] arrayRespuestas;
        RespuestasTest objetoResp;

        arrayRespuestas = new char[10];
        arrayRespuestas [0] = 'A';
        arrayRespuestas [1] = 'C';
        arrayRespuestas [2] = 'E';
        arrayRespuestas [3] = 'A';
        arrayRespuestas [4] = 'A';
        arrayRespuestas [5] = 'C';
        arrayRespuestas [6] = 'D';
        arrayRespuestas [7] = 'E';
        arrayRespuestas [8] = 'A';
        arrayRespuestas [9] = 'A';
        objetoResp = new RespuestasTest();
        objetoResp.escribirNumero (10);
        objetoResp.escribirArray (arrayRespuestas);
        flujoSalidaObjeto.writeObject (objetoResp);

        arrayRespuestas = new char[10];
        arrayRespuestas [0] = 'A';
        arrayRespuestas [1] = 'B';
        arrayRespuestas [2] = 'B';
        arrayRespuestas [3] = 'B';
        arrayRespuestas [4] = 'D';
        arrayRespuestas [5] = 'D';
        arrayRespuestas [6] = 'E';
        arrayRespuestas [7] = 'E';
        arrayRespuestas [8] = 'C';
        arrayRespuestas [9] = 'A';
        objetoResp = new RespuestasTest();
        objetoResp.escribirNumero (20);
        objetoResp.escribirArray (arrayRespuestas);
        flujoSalidaObjeto.writeObject (objetoResp);

        arrayRespuestas = new char [10];
        arrayRespuestas [0] = 'E';
        arrayRespuestas [1] = 'E';
        arrayRespuestas [2] = 'E';
        arrayRespuestas [3] = 'A';
        arrayRespuestas [4] = 'E';
        arrayRespuestas [5] = 'D';
        arrayRespuestas [6] = 'E';
        arrayRespuestas [7] = 'A';
        arrayRespuestas [8] = 'B';
        arrayRespuestas [9] = 'A';
        objetoResp = new RespuestasTest();
        objetoResp.escribirNumero (30);
        objetoResp.escribirArray (arrayRespuestas);
    }
}

```

```

        flujoSalidaObjeto.writeObject (objetoResp);

        arrayRespuestas = new char [10];
        arrayRespuestas [0] = 'C';
        arrayRespuestas [1] = 'E';
        arrayRespuestas [2] = 'B';
        arrayRespuestas [3] = 'B';
        arrayRespuestas [4] = 'C';
        arrayRespuestas [5] = 'D';
        arrayRespuestas [6] = 'A';
        arrayRespuestas [7] = 'C';
        arrayRespuestas [8] = 'C';
        arrayRespuestas [9] = 'D';
        objetoResp = new RespuestasTest ();
        objetoResp.escribirNumero (40);
        objetoResp.escribirArray (arrayRespuestas);
        flujoSalidaObjeto.writeObject (objetoResp);

        flujoSalidaObjeto.close ();

        System.out.println("Fin - fichero * respCorrectas.datos *");
    }
}

```

2. Realizar la codificación del segundo programa: CorreccionTest.java y completar la clase RespuestasTest.java. Para ello, se disponen de los siguientes ficheros:

- Fichero de entrada 1 de texto: “respuestas.txt”, que tiene el siguiente formato:

12345678	→ número de DNI
10	→ número de test que le ha correspondido
ABCAA_DEAA	→ respuestas a las 10 preguntas del test
87654321	
20	
ABBBDC__AB	→ 2 respuestas en blanco
12121212	
40	
BBCCDAECA_	→ 1 respuesta en blanco
12344321	
30	
EEEEAEDEABA	

Tener en cuenta que todas las respuestas proporcionadas por cada DNI, para un test, aparecen en la misma línea y también que puede haber respuestas en blanco.

- Fichero de entrada 2 de objetos: “respCorrectas.datos”, según el formato definido en la clase RespuestasTest.
- Fichero de salida de texto: “resultados.txt”, que tiene el siguiente formato:

DNI: 12345678	Puntos: 5.75	APTO
DNI: 87654321	Puntos: 3.0	NO APTO
DNI: 12121212	Puntos: -3.25	NO APTO
DNI: 12344321	Puntos: 10.0	APTO

La puntuación (tipo float) se obtiene de la siguiente forma:

- ☞ Respuesta correcta: sumar 1 punto.

➤ Respuesta en blanco: restar 0.25 puntos.

➤ Respuesta incorrecta: restar 0.50 puntos.

Si la puntuación es superior o igual a 5 puntos, es APTO; en caso contrario, es NO APTO.

Para realizar el segundo apartado debes tener en cuenta los siguientes pasos:

- a. Volcar el archivo “respCorrectas.datos” en un array de objetos de la clase RespuestasTest, gestionando todas las excepciones que se puedan producir y que estén relacionadas con archivos de objetos.
- b. Leer secuencialmente el fichero “respuestas.txt”, calculando para cada DNI la puntuación conseguida. Se supone que todos los datos de este fichero son correctos.
- c. Añadir los métodos (por ejemplo, de lectura de sus datos miembro) que sean necesarios en la clase RespuestasTest.

SOLUCIÓN

```
import java.io.*;

/**
 * La clase RespuestasTest
 */

class RespuestasTest implements Serializable
{
    //-----
    //    Datos Miembro
    //-----

    private int    numeroTest;
    private char [] arrayResp;

    //-----
    //    Constructores
    //-----
    public void RespuestasTest ()
    {
        numeroTest    = 0;
        arrayResp     = new char [10];
    }

    //-----
    //    Métodos Públicos
    //-----
    public void escribirNumero (int numero)
    {
        this.numeroTest = numero;
    }

    public void escribirArray (char [] array)
    {
        this.arrayResp = array;
    }

    public int leerNumero( )
    {
```

```

        return numeroTest;
    }

    public char [] leerArray( )
    {
        return arrayResp;
    }

    public char leerElementoArray(int i)
    {
        return arrayResp[i];
    }
}

/**
 * Programa CorreccionTest.java
 *
 **/

import java.io.*;

public class CorreccionTest
{
    public static void main(String[] args) throws IOException
    {
        //ficheros de entrada y de salida
        File ficheroEntrada1 = new File("respuestas.txt");
        FileReader leeFichero1 = new
        FileReader(ficheroEntrada1);
        BufferedReader lectorBuffer = new
        BufferedReader(leeFichero1);

        File ficheroEntrada2 = new File("respCorrectas.datos");
        FileInputStream flujoEntradaF2 = new
        FileInputStream(ficheroEntrada2);
        ObjectInputStream flujoEntradaObjeto = new
        ObjectInputStream(flujoEntradaF2);

        File ficheroSalida = new File("resultados.txt");
        FileOutputStream flujoFicheroSalida = new
        FileOutputStream(ficheroSalida);
        PrintWriter flujoSalida = new
        PrintWriter(flujoFicheroSalida);

        //variables relacionadas con los atributos
        //del fichero 'respuestas.txt'
        String numeroDNI, numeroTest, respTodas;
        float totalPuntos;
        char respCorrecta;

        //Fichero de entrada 'respCorrectas.datos'
        //
        //Cargar el archivo en un array de objetos RespuestasTest
        RespuestasTest objetoResp;

        RespuestasTest [] arrayResp = new RespuestasTest [4];

        int i = 0;
        boolean eof = false;

```

```

while(!eof)
{
    try
    {
        objetoResp = (RespuestasTest)
                    flujoEntradaObjeto.readObject();
        arrayResp [i] = objetoResp;
        i++;
    }
    catch(EOFException e)
    {
        eof = true;
    }
    catch (ClassNotFoundException e)
    {
        eof = true;
    }
}

flujoEntradaObjeto.close();

//Fichero de Entrada 'respuestas.txt'
//
//leer el String correspondiente al nombre del empleado
numeroDNI = lectorBuffer.readLine();

while(numeroDNI != null)
{
    totalPuntos = 0.0f;

    numeroTest = lectorBuffer.readLine();
    int n = Integer.parseInt(numeroTest);

    i = 0;
    while ( arrayResp[i].leerNumero() != n &&
            i < 4)
    {
        i++;
        //System.out.println("Valor de i: " + i);
    }
    respTodas = lectorBuffer.readLine();
    for (int j = 0; j < 10; j++)
    {
        respCorrecta = arrayResp[i].leerElementoArray(j);
        //System.out.println("Respuesta: " + respCorrecta);
        if (respTodas.charAt(j) == respCorrecta)
        {
            totalPuntos += 1;
        }
        else if (respTodas.charAt(j) == ' ')
        {
            totalPuntos -= 0.25;
        }
        else
        {
            totalPuntos -= 0.50;
        }
    }

    if (totalPuntos >= 5.0)
        flujoSalida.println("DNI: " + numeroDNI + "\t Puntos: " +

```

```
                totalPuntos+ "\t APTO" );
else
    flujoSalida.println("DNI: " + numeroDNI + "\t Puntos: " +
        totalPuntos+ "\t NO APTO" );

    numeroDNI = lectorBuffer.readLine();
}

System.out.println("Fin - fichero * resultados.txt *");

//cerrar los ficheros
lectorBuffer.close();
flujoSalida.close();
}
}
```