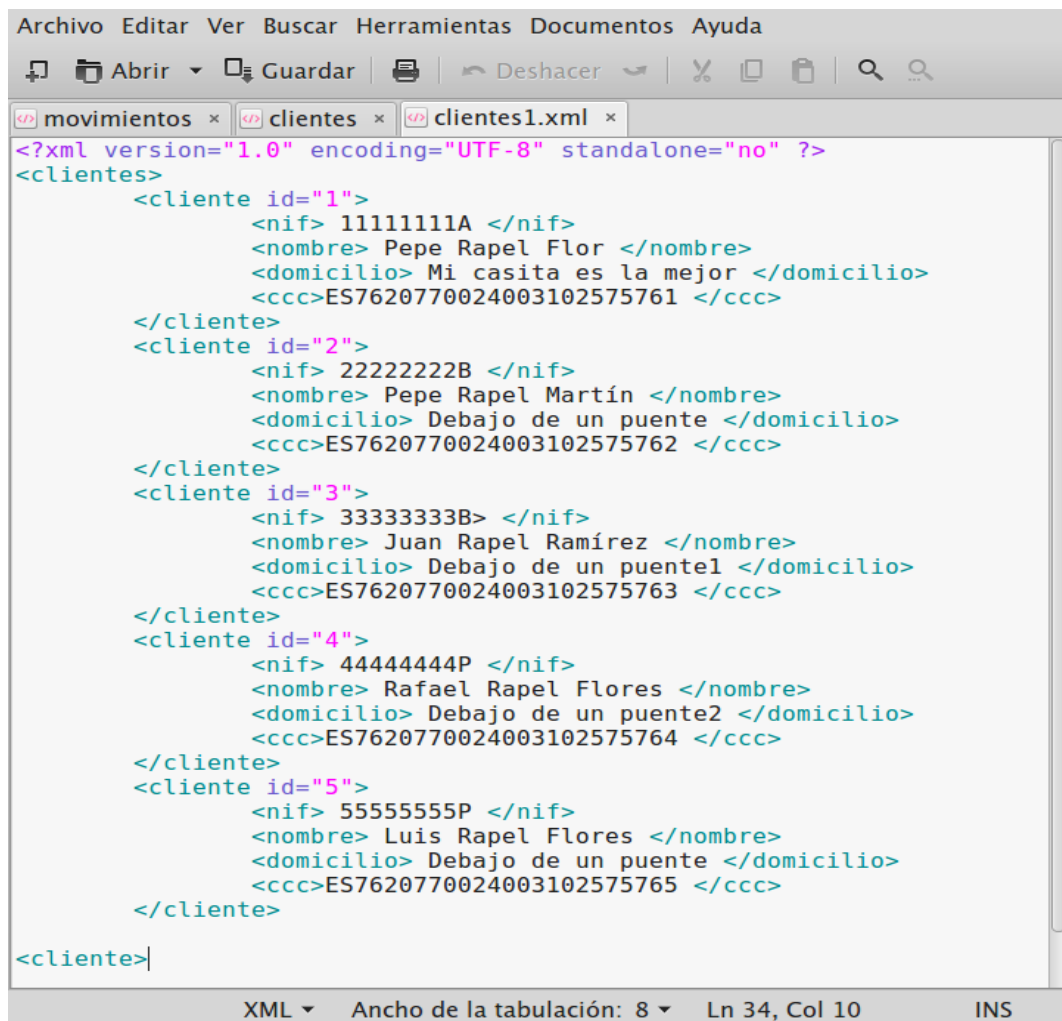


Desarrolla un programa que va a llevar la gestión de cuentas de los clientes del Banco NOMEPIDASUNDURO.

Llevaremos la gestión de clientes de la siguiente forma.

1) Debemos desarrollar una aplicación y Clase llamada **ImportarClientes** que lea **a partir de un directorio pasado como argumento** una serie de **ficheros en formato XML** con la información de los clientes a dar de alta.

Ejemplo del fichero clientes1.xml



```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<clientes>
  <cliente id="1">
    <nif> 11111111A </nif>
    <nombre> Pepe Rapel Flor </nombre>
    <domicilio> Mi casita es la mejor </domicilio>
    <ccc>ES7620770024003102575761 </ccc>
  </cliente>
  <cliente id="2">
    <nif> 22222222B </nif>
    <nombre> Pepe Rapel Martín </nombre>
    <domicilio> Debajo de un puente </domicilio>
    <ccc>ES7620770024003102575762 </ccc>
  </cliente>
  <cliente id="3">
    <nif> 33333333B </nif>
    <nombre> Juan Rapel Ramírez </nombre>
    <domicilio> Debajo de un puentel </domicilio>
    <ccc>ES7620770024003102575763 </ccc>
  </cliente>
  <cliente id="4">
    <nif> 44444444P </nif>
    <nombre> Rafael Rapel Flores </nombre>
    <domicilio> Debajo de un puente2 </domicilio>
    <ccc>ES7620770024003102575764 </ccc>
  </cliente>
  <cliente id="5">
    <nif> 55555555P </nif>
    <nombre> Luis Rapel Flores </nombre>
    <domicilio> Debajo de un puente </domicilio>
    <ccc>ES7620770024003102575765 </ccc>
  </cliente>
</cliente>
```

En caso de que exista el cliente (con el mismo nif) en nuestra base de datos actualizaremos sus datos.

Como resultado, se debe crear, si no existe, un fichero de texto llamado **clientes.txt** en un directorio **pasado también como argumento**, en donde aparezca en cada línea la información de cada cliente. El campo final de cada línea es su saldo global. Al darse de alta tendrá un saldo de 0,00

Ejemplo del contenido del fichero.

Archivo Editar Ver Buscar Herramientas Documentos Ayuda

  Abrir ▾  Guardar |  |  Deshacer  |    |  

 movimientos ×  clientes ×  clientes1.xml ×  clientes.txt ×

```
11111111A%Pepe Rapel Flor%Mi casita es la mejor%ES7620770024003102575761%15267,91
22222222B%Pepe Rapel Martín%Debajo de un puente%ES7620770024003102575762%607,00
33333333B%Juan Rapel Ramírez%Debajo de un puente1%ES7620770024003102575763%0,00
44444444P%Rafael Rapel Flores%Debajo de un puente2%ES7620770024003102575764%0,00
55555555P%Luis Rapel Flores%Debajo de un puente%ES7620770024003102575765%0,00|
```

Texto plano ▾

Ancho de la tabulación: 8 ▾

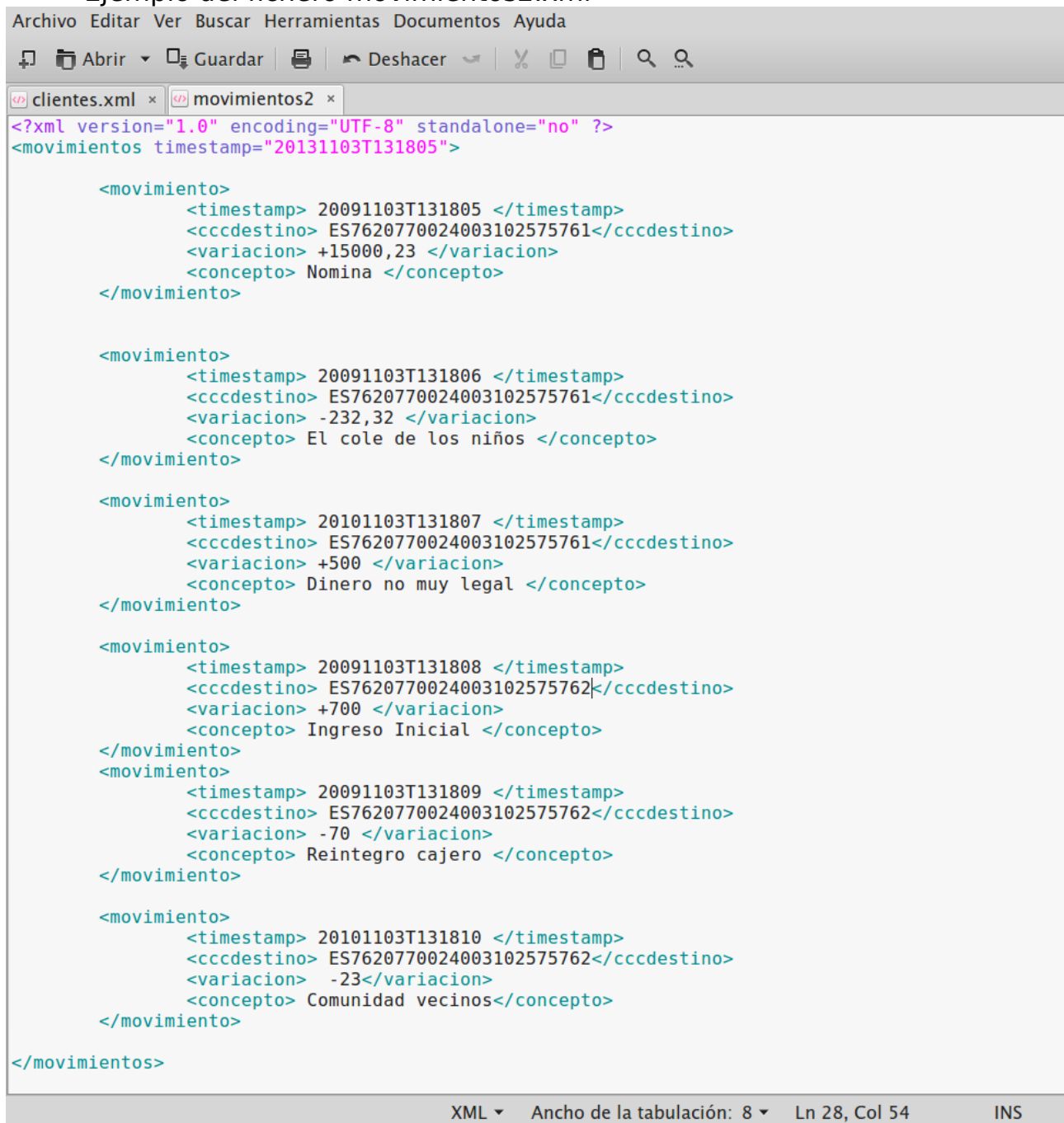
Ln 5, Col 78

INS

2)

a) Debemos desarrollar una aplicación y Clase llamada **ImportarMovimientos** que tenga como primer argumento un **directorio**. En ese directorio **debe** haber una serie de **ficheros en formato XML** con la información de los movimientos sobre las cuentas de los usuarios.

Ejemplo del fichero movimientos2.xml



```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<movimientos timestamp="20131103T131805">

  <movimiento>
    <timestamp> 20091103T131805 </timestamp>
    <cccdestino> ES7620770024003102575761</cccdestino>
    <variacion> +15000,23 </variacion>
    <concepto> Nomina </concepto>
  </movimiento>

  <movimiento>
    <timestamp> 20091103T131806 </timestamp>
    <cccdestino> ES7620770024003102575761</cccdestino>
    <variacion> -232,32 </variacion>
    <concepto> El cole de los niños </concepto>
  </movimiento>

  <movimiento>
    <timestamp> 20101103T131807 </timestamp>
    <cccdestino> ES7620770024003102575761</cccdestino>
    <variacion> +500 </variacion>
    <concepto> Dinero no muy legal </concepto>
  </movimiento>

  <movimiento>
    <timestamp> 20091103T131808 </timestamp>
    <cccdestino> ES7620770024003102575762</cccdestino>
    <variacion> +700 </variacion>
    <concepto> Ingreso Inicial </concepto>
  </movimiento>
  <movimiento>
    <timestamp> 20091103T131809 </timestamp>
    <cccdestino> ES7620770024003102575762</cccdestino>
    <variacion> -70 </variacion>
    <concepto> Reintegro cajero </concepto>
  </movimiento>

  <movimiento>
    <timestamp> 20101103T131810 </timestamp>
    <cccdestino> ES7620770024003102575762</cccdestino>
    <variacion> -23</variacion>
    <concepto> Comunidad vecinos</concepto>
  </movimiento>

</movimientos>
```

Cada movimiento contiene la información siguiente:

<timestamp> → La hora y día del movimiento hacia la cuenta del usuario. Está expresada en notación ISO_8601

<ccdestino> → La cuenta en la que se realiza el movimiento.
<variacion> → + implica un aumento de saldo y - implica una disminución de saldo
<concepto> → Concepto del movimiento.

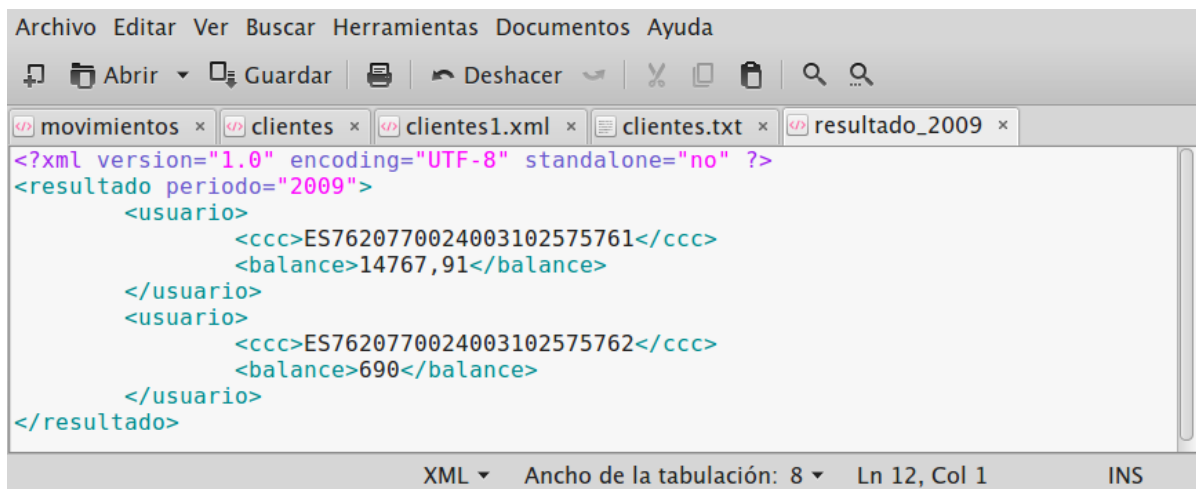
La aplicación deberá almacenar dicha información en **ficheros binarios en un directorio que también se pasará como argumento**. Además se deberá actualizar la información de saldo global que aparece en **clientes.txt**. **La dirección de este fichero también se pasará como argumento de esta aplicación.**

b) Debemos desarrollar una aplicación y Clase llamada **ObtenerBalanceDelAño** que tome como parámetros:

- 1) El año a mostrar
- 2) La dirección del directorio con los ficheros binarios en donde se guardará la base de datos de los movimientos en formato que creas oportuno.
- 3) La dirección del directorio en donde se va a crear el fichero en XML que resuma los movimientos de ese año.

El fichero resumen estará en formato XML y debe tener un balance en ese año, por cada cuenta que tuvo movimiento/s en ese año.

Ejemplo del fichero XML para el año 2009 del ejemplo:



The screenshot shows a text editor window with the following menu: Archivo, Editar, Ver, Buscar, Herramientas, Documentos, Ayuda. The toolbar includes icons for opening, saving, printing, undo, redo, cut, copy, paste, and search. The tabs at the top are: movimientos, clientes, clientes1.xml, clientes.txt, and resultado_2009. The main text area contains the following XML code:

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<resultado periodo="2009">
  <usuario>
    <ccc>ES7620770024003102575761</ccc>
    <balance>14767,91</balance>
  </usuario>
  <usuario>
    <ccc>ES7620770024003102575762</ccc>
    <balance>690</balance>
  </usuario>
</resultado>
```

The status bar at the bottom indicates: XML, Ancho de la tabulación: 8, Ln 12, Col 1, and INS.

Criterios de evaluación

- Se han utilizado clases para la gestión de ficheros y directorios.
- Se han valorado las ventajas y los inconvenientes de las distintas formas de acceso.
- Se han utilizado clases para recuperar información almacenada en un fichero XML.
- Se han utilizado clases para almacenar información en un fichero XML.
- Se han utilizado clases para convertir a otro formato información contenida en un fichero XML.
- Se han previsto y gestionado las excepciones.
- Se han probado y documentado las aplicaciones desarrolladas.

Criterios de calificación.

Esta prueba se corresponde con el 50% de la calificación de esta unidad. Hay que obtener un 4 como mínimo en esta prueba para poder hacer media con las partes de ejercicios, trabajos y actitud de esta misma unidad.

Se tendrá en cuenta:

- 1) Esta prueba dura 3 horas en total.
- 2) La prueba será realizada de forma **personal e individual** con el uso del ordenador y de Internet.

Los ejercicios están ponderados de la siguiente manera:

Ejercicios

- 1) 30%
- 2.a) 35%
- 2.b) 45%

3) Se tendrá en cuenta **en cada aplicación**:

- a) Que el funcionamiento es el correcto según lo especificado. (90% de la calificación)
- b) Se han utilizado las clases y la forma de acceso más oportunas y convenientes según el tipo de problema planteado. (5% de la calificación)
- c) Se hacen comentarios no oportunos y superfluos en el código o no se hacen comentarios. (-5% de la calificación)
- d) Se gestionan las excepciones de forma correcta. (5% de la calificación)
- e) Se realizará las siguientes entregas **mínimas** en Moodle:
 - El proyecto o proyecto en Eclipse de cada aplicación que permita al profesor su corrección desde los fuentes.
 - El fichero jar con las clases, y con la clase con main puesto en el Manifest: para poder ser ejecutada desde un terminal.
Ej: *java -jar **ImportarClientes.jar** dir1 dir2*
 - Un **script** que permita ejecutar nuestra aplicación. En él se definirá correctamente el **CLASSPATH(si hiciera falta)** **para poder ejecutar nuestra aplicación.**

