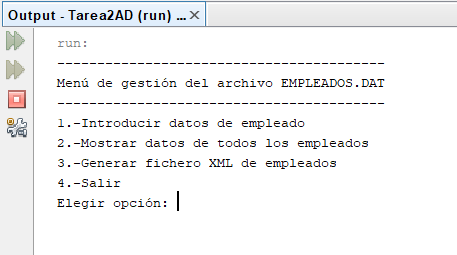
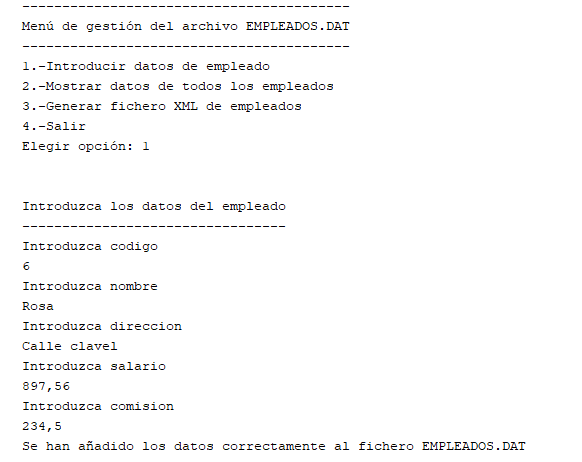
**APARTADO 1**

Se nos pide crear una aplicación para generar un fichero de EMPLEADOS.DAT que contenga los datos de los empleados y basándonos en ese fichero creemos el xml con la clase DOM de la API de java.

Para ello he creado 3 clases una para empleados que contendrá los datos del mismo, una clase para los métodos para la gestión de entrada/salida de datos del fichero y una clase principal donde mostraremos un menú con 4 opciones:

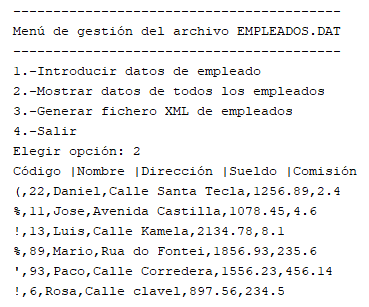


1. Ingresar nuevo empleado y añadirlo al fichero o crearlo si no existe.

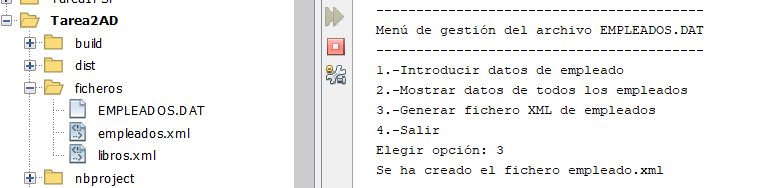


*La aplicación solicita los datos de empleado, los comprueba y los añade al fichero mostrándonos por pantalla un mensaje de confirmación*

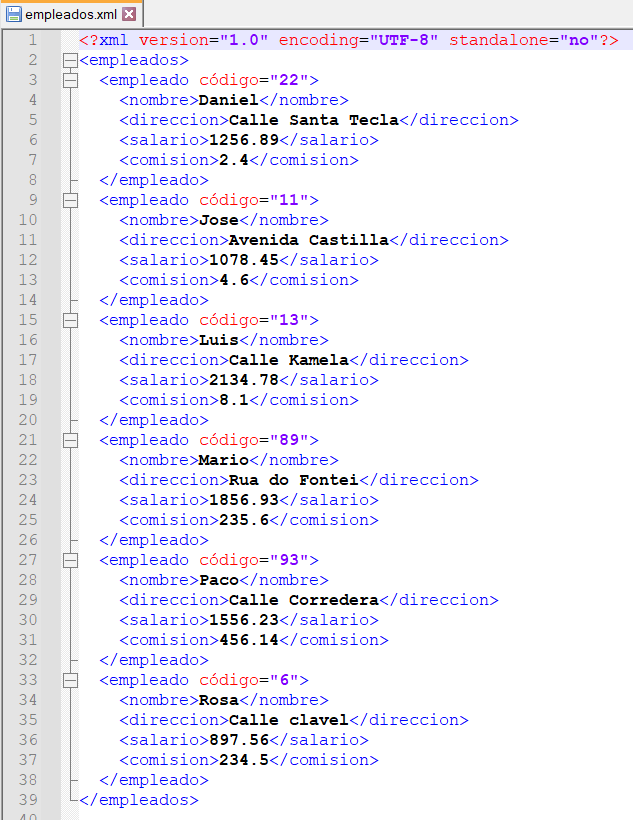
1. La segunda es mostrar el contenido del fichero por pantalla, aunque no lo pide la práctica nos sirve para visualizar el contenido sin tener que abrir el fichero.

  
*Podemos ver el contenido del fichero que antes de introducir el último empleado ya contenía 5 empleados como nos pedía la tarea*

1. La tercera genera el fichero empleados.xml leyendo el con el contenido del fichero EMPLEADOS.DAT



*Podemos ver a la izquierda que se ha generado el fichero mostrando por pantalla un mensaje de confirmación.*



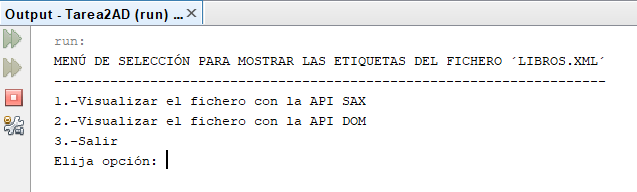
*Este sería el fichero xml que nos genera la aplicación haciendo uso de DOM como nos pide la tarea*

1. La opción Salir cierra la aplicación.

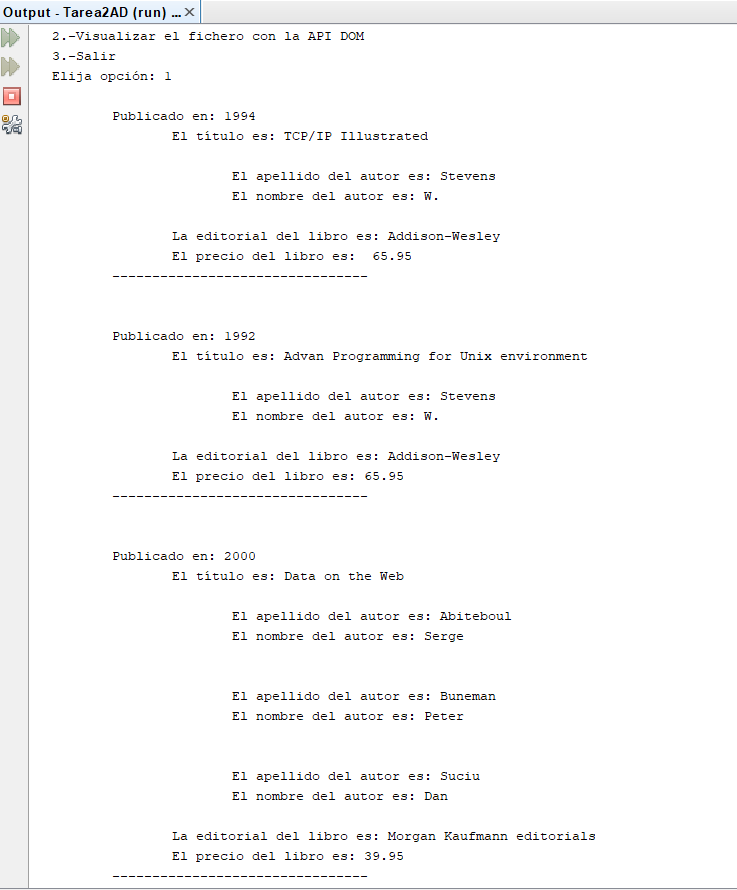
**APARTADO 2**

Aquí nos piden que mostremos por pantalla el fichero “libros.xml” que nos facilita la propia tarea utilizando DOM y SAX de la API de java.

Para ello creamos una clase libros que contendrá los datos del mismo, un manejador de archivos necesario para parsear con SAX y una clase principal o main donde mostramos al usuario un menú con las 2 opciones de visualización para que sea el quien seleccione la opción:

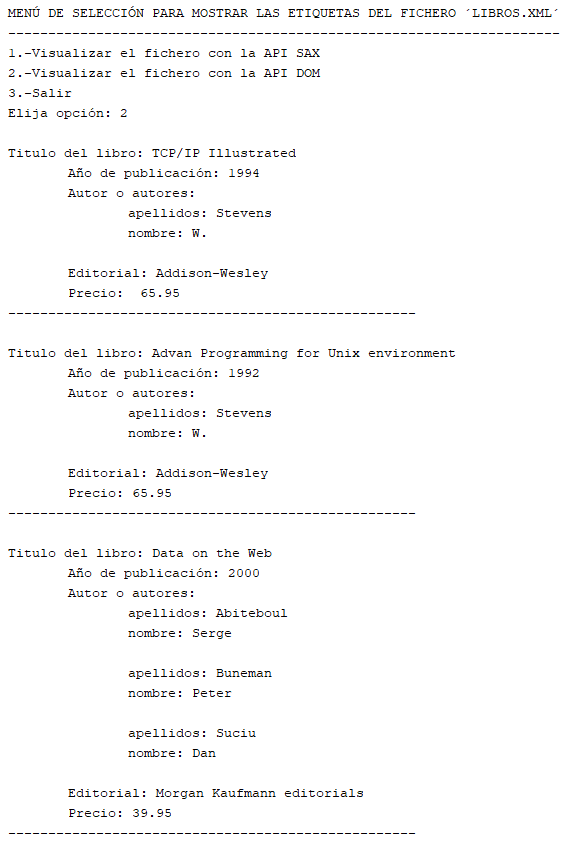


1. Visualizar con SAX el fichero



*Esta sería el resultado de la visualización con SAX, normalizado y tabulado*

1. Visualizar con DOM el fichero



*Esta sería el resultado de la visualización con DOM, normalizado y tabulado*

En conclusión, como podemos ver el resultado de la visualización es prácticamente el mismo, la diferencia principal entre los 2 está en la forma en que leen el fichero, SAX lo hace secuencialmente y DOM va leyendo por nodos que son los elementos y atributos del xml. De cara al usuario el resultado es el mismo porque la diferencia está por detrás, con clases y métodos distintos.