

Práctica 4: Explotación de una Base de Datos Utilizando *Java Persistence API (JPA)*

Bases de Datos 2

Grado en Ingeniería Informática
Departamento de Informática e Ingeniería de Sistemas
Universidad de Zaragoza

1.- Objetivos

En esta práctica el alumnado utilizará *Java Persistence API (JPA)*, y concretamente *Hibernate* como *middleware*, para realizar un *mapeo* objeto/relacional.

En mayor detalle, los objetivos de la práctica son:

1. Generar automáticamente un esquema relacional a partir de un esquema de clases en Java.
 - Para ello, definiremos en Java las clases necesarias para modelar el problema planteado en las primeras prácticas de la asignatura (base de datos de *banquito*). Luego, añadiremos a las clases Java las anotaciones JPA necesarias para definir el *mapeo* objeto/relacional deseado. Por último, utilizaremos *Hibernate* para generar automáticamente el esquema de base de datos relacional correspondiente.
2. Establecer las correspondencias entre un esquema de clases Java y un esquema relacional existente.
 - Para ello, definiremos las clases Java con las anotaciones JPA necesarias para establecer un *mapeo* objeto/relacional que permita establecer las correspondencias entre el esquema relacional planteado en la primera práctica de la asignatura y el esquema orientado a objetos definido en Java. En relación con este objetivo, debe destacarse que no debe modificarse el esquema original de la base de datos definido previamente.
 - Como alternativa, puede utilizarse una herramienta de ingeniería inversa (por ejemplo, Eclipse con la faceta JPA, como se describe en el documento “Ingeniería inversa JPA” disponible en el sitio Moodle de la asignatura). En este caso, es necesario analizar y comentar los resultados de la ingeniería inversa automática, no simplemente confiar en que todas las transformaciones sean las más adecuadas.
3. Formular consultas de ejemplo utilizando alguna o varias de las siguientes opciones: *JPQL*, *Criteria API*, y *SQL nativo*.

2.- Contexto del Problema

En la primera práctica de la asignatura hemos abordado el diseño de una base de datos relacional simplificada para un banco. En esta práctica abordaremos la relación existente entre el esquema relacional de la base de datos del banco y las clases de aplicación que configuran un esquema orientado a objetos. Se analizarán dos opciones diferentes:

1. *Generación automática del esquema de BD.* Definir el esquema de clases en Java y, tras establecer las anotaciones JPA necesarias, generar automáticamente un esquema de base de datos relacional a partir del esquema de clases.
2. *Esquema de BD preexistente.* Definir tanto el esquema de clases en Java como el esquema relacional de la base de datos (ya obtenido en las prácticas previas) y establecer las anotaciones JPA necesarias para establecer la correspondencia entre ambos.

Se considerará el SGBD Oracle. Opcionalmente, puede probarse también con otros SGBD (destacándolo y explicándolo adecuadamente en la memoria de prácticas).

La tercera (y última) parte de la práctica implica proponer al menos 5 consultas de ejemplo interesantes (no triviales) y formularlas utilizando alguna o varias de las 3 opciones siguientes (se recomienda no limitarse a SQL nativo, probando en la medida de lo posible varias opciones y comparándolas):

1. JPQL
2. Criteria API
3. SQL nativo

Los metadatos JPA se embeberán en el código de las clases Java. Adicionalmente, de forma opcional, puede considerarse también la posibilidad de evitar tener que incluir dichos metadatos embebidos en los ficheros fuente, mediante la utilización exclusiva de ficheros de configuración XML. En caso de considerar también esta posibilidad, se recomienda comentarla y analizarla en la memoria de prácticas entregada, destacando que también se ha analizado y probado esta opción.

Aunque la práctica se realizará con *Hibernate*, no deben utilizarse características propias de *Hibernate* que no estén contempladas en la especificación de JPA. No obstante, pueden explotarse características avanzadas siempre que sea como añadidos extra opcionales (no afectando al núcleo de la práctica y separándolos y documentándolos adecuadamente). Si se identifican funcionalidades relevantes de *Hibernate* que no formen parte del estándar, pueden comentarse y analizarse en la memoria de prácticas.

3.- Aclaraciones

Es conveniente realizar las siguientes aclaraciones:

- Es importante extraer conclusiones: resultados obtenidos, indagaciones realizadas, dificultades encontradas, valoración de los aprendizajes o mejoras de habilidades conseguidas, opinión personal, etc.
 - En particular, para esta práctica es relevante destacar aquellos aspectos que hayáis averiguado por vosotros/as mismos/as (no tratados directamente en clase), así como las dificultades encontradas. Asimismo, se incluirán explicaciones didácticas, paso a paso (tipo tutorial), donde esto pueda resultar relevante. Además, en el caso de la generación automática del esquema de bases de datos relacional, se debe analizar y comentar el esquema generado.

- Cuando resulte relevante, el profesorado proporcionará aclaraciones adicionales y ejemplos para el alumnado a través de Moodle.

4.- Entrega de la Práctica

Se debe entregar un fichero zip denominado p4-<nip>.zip (donde <nip> representa el NIP del coordinador/responsable del grupo) con el siguiente contenido:

1. Fichero autores.txt: contendrá el nombre y apellido de los autores de la práctica y sus NIP.
2. Directorio *fuentes*: contendrá todo el código desarrollado para la práctica:
 - a. Fichero java-gen-esquema.zip: fuentes Java anotados para la generación automática del esquema relacional en Oracle, junto con los ficheros de configuración necesarios, así como scripts “compila” y “ejecuta” (tanto para Windows como para Unix/Linux), y cualquier otro fichero que pueda ser relevante. Debe incluirse también un fichero “Test.java” que permita fácilmente poblar la BD con datos de ejemplo y realizar las 5 preguntas de ejemplo propuestas.
 - b. Fichero java-previo-esquema.zip: fuentes Java anotados para establecer la correspondencia con el esquema relacional pre-existente definido en las prácticas previas, junto con los ficheros de configuración necesarios, así como scripts “compila” y “ejecuta” (tanto para Windows como para Unix/Linux), un fichero “sql.txt” con las sentencias SQL que se ejecutan para generar el esquema de la bases de datos previa, y cualquier otro fichero que pueda ser relevante. Debe incluirse también un fichero “Test.java” que permita fácilmente poblar la BD con datos de ejemplo y realizar las 5 preguntas de ejemplo propuestas. Es necesario comentar en la memoria las anotaciones relevantes, tanto si se han anotado manualmente los ficheros como si se ha utilizado alguna herramienta de ingeniería inversa.
Si se hubiera realizado también la parte correspondiente a la definición de metadatos en un fichero de configuración XML, se incluirá también un fichero equivalente java-previo-esquema-metadatos-XML.zip con los fuentes y ficheros de prueba correspondientes a dicha solución.

Importante: los scripts deberán poder ejecutarse sin problemas con independencia del estado inicial de la base de datos destino, de forma que en el script hay que asegurarse de limpiar adecuadamente el espacio de tablas de la base de datos para evitar conflictos con objetos del mismo nombre que pudieran existir previamente (tablas, restricciones, etc.): como ejemplo de prueba, debe ser posible ejecutar dos veces seguidas el script y no producirse errores.

3. Fichero memoria-p4.pdf, conteniendo la memoria de la práctica, de acuerdo con las instrucciones generales de elaboración de prácticas.

Instrucciones de envío:

El fichero anterior se someterá a través de *Moodle* (se habilitará una opción a tal efecto) utilizando la cuenta del coordinador/responsable del grupo.

Hay que asegurarse de que la práctica funciona correctamente y que será fácilmente reproducible (incluir ficheros de scripts para Windows y Linux). También es importante someter código limpio (donde se ha evitado introducir mensajes de depuración y comentarios que no proporcionan información relevante).

Fecha límite de entrega:

La fecha límite de entrega será el día anterior a la siguiente sesión de prácticas.