



AE-4. JPA

Acceso a Datos

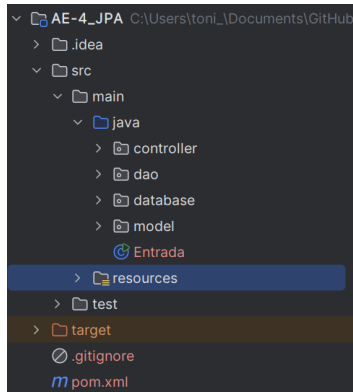
DOCUMENTACIÓN

Archivo documentado sobre la actividad AE-4 de la asignatura de Acceso a Datos.

Antonio Martinez Segura
2º DAM

En este documento presentaré el desarrollo de la actividad realizada documentando las partes más significativas de éste.

El proyecto ha sido estructurado en paquetes para facilitar la organización y escalabilidad del código:



Model → es donde podemos encontrar las clases que representan las entidades de la base de datos. He utilizado la dependencia de lombok para crear los constructores y Getter/Setter. Cada clase tiene sus atributos correspondientes y las diferentes anotaciones de hibernate.

Dao → En este paquete he implementado los métodos para la base de datos, habrá un dao para cada entidad que precise de métodos.

Controller → Es el paquete donde se aplica la lógica de negocio.

Database → Paquete donde a través de HibernateUtil se administra la SessionFactory. Abre y cierra la sesión.

Resources → Paquete donde se encuentra el archivo hibernate.cfg.xml. En este archivo se configuran los detalles de conexión(URL, usuario, password, dialect..), además es donde se realiza el mapeo de cada clase del paquete model.

Para poder llevar a cabo la actividad de manera completa, he tenido que crear una base de datos llamada “librerías” y crear todas las entidades que se pedían.

▼	librerias	
▼	Tables	
>	autores	16K
>	editoriales	32K
>	jefes	16K
>	librerias	32K
>	librerias_libros	48K
>	libros	48K
>	pedidos	32K
>	clientes	16K
>	libreria_cliente	48K

La actividad precisaba de crear varias tablas “renacidas” para las relaciones N / M(de muchos a muchos).

Por último, la clase Entrada sirve como punto de entrada de la aplicación. Tiene como finalidad coordinar las llamadas al controller y ejecutar el programa.

Resultados:

- Dar de alta 3 autores.

```
//Autor
controlador.crearAutor(new Autor( nombre: "Elisabet", apellidos: "Benavent", fechaNac: "12/08/1987"));
controlador.crearAutor(new Autor( nombre: "Bryan", apellidos: "Weis", fechaNac: "15/06/1975"));
controlador.crearAutor(new Autor( nombre: "Luis", apellidos: "Perez", fechaNac: "18/01/1990"));
```

123 id	A-Z nombre	A-Z apellidos	A-Z fecha_nac
1	Elisabet	Benavent	12/08/1987
2	Bryan	Weis	15/06/1975
3	Luis	Perez	18/01/1990

- Dar de alta 2 editoriales.

```
//Editorial
controlador.crearEditorial(new Editorial( nombre: "ANAYA", direccion: "Calle la Luna"));
controlador.crearEditorial(new Editorial( nombre: "PLANETA", direccion: "Calle los Astros"));
```

id	Az nombre	Az direccion
1	ANAYA	Calle la Luna
2	PLANETA	Calle los Astros

- Dar de alta 8 libros, cada libro será escrito por uno de los autores dados de alta previamente y pertenecerá a uno de los editoriales dados de alta previamente.

```
//Libros
controlador.crearLibro(new Libro( titulo: "Mi isla", precio: 22.90), idAutor: 1, idEditorial: 1);
controlador.crearLibro(new Libro( titulo: "Muchas vidas, muchos Maestros", precio: 19.90), idAutor: 2, idEditorial: 2);
controlador.crearLibro(new Libro( titulo: "La Rueda", precio: 21.20), idAutor: 3, idEditorial: 1);
controlador.crearLibro(new Libro( titulo: "La Noche", precio: 25.99), idAutor: 1, idEditorial: 2);
controlador.crearLibro(new Libro( titulo: "Siempre a tu lado", precio: 20.90), idAutor: 1, idEditorial: 1);
controlador.crearLibro(new Libro( titulo: "Amanecer", precio: 17.99), idAutor: 2, idEditorial: 2);
controlador.crearLibro(new Libro( titulo: "Noche oscura", precio: 12.90), idAutor: 3, idEditorial: 1);
controlador.crearLibro(new Libro( titulo: "Siete vidas", precio: 18.90), idAutor: 1, idEditorial: 2);
```

id	Az titulo	123 precio	123 editorial_id	123 autor_id
1	Mi isla	22,9	1	1
2	Muchas vidas, muchos Maestros	19,9	2	2
3	La Rueda	21,2	1	3
4	La Noche	25,99	2	1
5	Siempre a tu lado	20,9	1	1
6	Amanecer	17,99	2	2
7	Noche oscura	12,9	1	3
8	Siete vidas	18,9	2	1

- 2 librerías, cada librería tendrá 4 libros dados de alta previamente.

```
//crear librerías y asignar libros

Libreria libreria1 = new Libreria( nombre: "La casa del libro", dueño: "Tomás Martínez", direccion: "Avenida Perez Galdos");
Libreria libreria2 = new Libreria( nombre: "Libreria Central", dueño: "Ana Martínez", direccion: "Calle Mayor");

//obtener los libros creados
List<Libro> libros = controlador.obtenerLibros();

//asignar los libros a las librerías
if (libros.size() >= 8) {
    libreria1.setLibros(libros.subList(0, 4));
    libreria2.setLibros(libros.subList(4, 8));

    //guardar las librerías

    controlador.crearLibreria(libreria1, libreria1.getLibros());
    controlador.crearLibreria(libreria2, libreria2.getLibros());
}
else {
    System.out.println("No hay suficientes libros en la base de datos.");
}
```

123 id	A-Z nombre	A-Z dueño	A-Z direccion
7	La casa del libro	Tomás Martínez	Avenida Perez Galdos
8	Librería Central	Ana Martínez	Calle Mayor

- Tabla Renacida librerías_libros

123 libreria_id	123 libro_id
7	1
7	2
7	3
7	4
8	5
8	6
8	7
8	8

- Mostrar todos los libros dados de alta, con su editorial y su autor

```
// Mostrar todos los libros con su autor y editorial
System.out.println("Libros por autor y editorial:\n");
mostrarLibrosConAutorYEditorial(controlador);
```

```
// mostrar todos los libros con su autor y editorial
public static void mostrarLibrosConAutorYEditorial(Controlador controlador) { 1usage
    List<Libro> libros = controlador.obtenerLibrosConAutorYEditorial();
    for (Libro libro : libros) {
        System.out.println("Titulo: " + libro.getTitulo());
        System.out.println("Autor: " + libro.getAutor().getNombre() + " " + libro.getAutor().getApellidos());
        System.out.println("Editorial: " + libro.getEditorial().getNombre());
        System.out.println("-----");
    }
}
```

```
Titulo: Mi isla
Autor: Elisabet Benavent
Editorial: ANAYA
-----
Titulo: Muchas vidas, muchos Maestros
Autor: Bryan Weis
Editorial: PLANETA
-----
Titulo: La Rueda
Autor: Luis Perez
Editorial: ANAYA
-----
Titulo: La Noche
Autor: Elisabet Benavent
Editorial: PLANETA
-----
Titulo: Siempre a tu lado
Autor: Elisabet Benavent
Editorial: ANAYA
-----
Titulo: Amanecer
Autor: Bryan Weis
Editorial: PLANETA
-----
Titulo: Noche oscura
Autor: Luis Perez
Editorial: ANAYA
-----
Titulo: Siete vidas
Autor: Elisabet Benavent
Editorial: PLANETA
-----
```

- Mostrar todos los autores dados de alta, con sus libros asociados

```
Autores y sus libros:

Autor: Elisabet Benavent
Libros:
- Mi isla
- La Noche
- Siempre a tu lado
- Siete vidas
-----
Autor: Bryan Weis
Libros:
- Muchas vidas, muchos Maestros
- Amanecer
-----
Autor: Luis Perez
Libros:
- La Rueda
- Noche oscura
-----
```

```
// mostrar todos los autores con sus libros
public static void mostrarAutoresConLibros(Controlador controlador) { 1usage
    List<Autor> autores = controlador.obtenerAutoresConLibros();
    for (Autor autor : autores) {
        System.out.println("Autor: " + autor.getNombre() + " " + autor.getApellidos());
        System.out.println("Libros: ");
        for (Libro libro : autor.getLibros()) {
            System.out.println("- " + libro.getTitulo());
        }
        System.out.println("-----");
    }
}
```

```
// Mostrar todos los autores con sus libros
System.out.println("Autores y sus libros:\n");
mostrarAutoresConLibros(controlador);
```

- Mostrar todas las librerías, con solamente sus libros asociados

```
// mostrar todas las librerías con sus libros
public static void mostrarLibreriasConLibros(Controlador controlador) {
    List<Libreria> librerias = controlador.obtenerLibreriasConLibros();
    for (Libreria libreria : librerias) {
        System.out.println("Librería: " + libreria.getNombre());
        System.out.println("Libros: ");
        for (Libro libro : libreria.getLibros()) {
            System.out.println(" - " + libro.getTitulo());
        }
        System.out.println("-----");
    }
}

// Mostrar todas las librerías con sus libros
System.out.println("Librerías y libros:\n");
mostrarLibreriasConLibros(controlador);
```

Librerías y libros:

Librería: La casa del libro
Libros:
- Mi isla
- Muchas vidas, muchos Maestros
- La Rueda
- La Noche

Librería: Librería Central
Libros:
- Siempre a tu lado
- Amanecer
- Noche oscura
- Siete vidas

- Mostrar todos los libros dados de alta, y en la librería en la que están.

```
Libros y librerías en las que se encuentran:

Título: Mi isla
Librerías:
- La casa del libro
-----
Título: Muchas vidas, muchos Maestros
Librerías:
- La casa del libro
-----
Título: La Rueda
Librerías:
- La casa del libro
-----
Título: La Noche
Librerías:
- La casa del libro
-----
Título: Siempre a tu lado
Librerías:
- Librería Central
-----
Título: Amanecer
Librerías:
- Librería Central
-----
Título: Noche oscura
Librerías:
- Librería Central
-----
Título: Siete vidas
Librerías:
- Librería Central
```

```
// mostrar todos los libros con las librerías en las que están
public static void mostrarLibrosConLibrerias(Controlador controlador) {
    List<Libro> libros = controlador.obtenerLibrosConLibrerias();
    for (Libro libro : libros) {
        System.out.println("Título: " + libro.getTitulo());
        System.out.println("Librerías: ");
        for (Libreria libreria : libro.getLibrerias()) {
            System.out.println(" - " + libreria.getNombre());
        }
        System.out.println("-----");
    }
}

// Mostrar todos los libros con las librerías en las que están
System.out.println("Libros y librerías en las que se encuentran:\n");
mostrarLibrosConLibrerias(controlador);
```

Requerimiento 2

He creado la clase “Jefe” para la relación OneToOne y en la base de datos la entidad “jefes”

```
@Entity
@Table(name = "jefes")
public class Jefe implements Serializable {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;

    @Column
    private String nombre;

    @OneToOne(mappedBy = "jefe")
    private Libreria libreria;
```

He creado la clase “Pedido” para la relación OneToMany y la entidad “pedidos” para la base de datos.

```
@Entity
@Table(name = "pedidos")
public class Pedido implements Serializable {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;

    @Column
    private String estado;

    @ManyToOne
    @JoinColumn(name = "libreria_id")
    private Libreria libreria;
```

He creado la clase “Cliente” para la relación ManyToMany y la entidad “clientes” para la base de datos.

```
@Entity
@Table(name = "clientes")
public class Cliente implements Serializable {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;

    @Column
    private String nombre;

    @Column
    private String telefono;

    @ManyToMany(mappedBy = "clientes") // Relación inversa
    private List<Libreria> librerias;
```

Las tres clases se relacionan directamente con librería y además “Cliente” y “Librería” tiene una tabla renacida “libreria_cliente” dada su relación N/M.

```
@OneToOne
@JoinColumn(name = "jefe_id")
private Jefe jefe;

@OneToMany(mappedBy = "libreria")
private List<Pedido> pedidos;

@ManyToMany
@JoinTable(
    name = "libreria_cliente",
    joinColumns = @JoinColumn(name = "libreria_id"),
    inverseJoinColumns = @JoinColumn(name = "cliente_id")
)
private List<Cliente> clientes;
```