


Evaluación: 1 – UT2 Conexiones a BD Relacionales	Fecha: 25/10/2024	Calificación	
Ciclo: DAM	Curso: 2ºA		
Módulo: ACDA – Acceso a Datos	Dpto.: INFORMÁTICA		
Nombre y apellidos:			

No dejes que lo que no puedes hacer interfiera con lo que puedes hacer (John R. Wooden)

Para la realización de este examen **descarga de Moodle** el proyecto UT2_EXAMEN, **descomprímelo y ábrelo desde IntelliJ**.

IMPORTANTE:

- Proyecto entregado con errores de compilación **no se corrige**
 - Las líneas comentadas **no se tendrán en cuenta**
 - Todo código hecho fuera de lugar **no se tendrá en cuenta**. Hacer todo el código entre las etiquetas TODO Start y TODO End (salvo en el apartado 1 que no se especifica qué parte de la clase tienes que editar)
1. **Arregla** la clase **MySQLConnection** para que **conecte con tu BD**
 2. **Define** los atributos y métodos necesarios de la **clase modelo Historia** teniendo en cuenta que tiene los atributos **escritos exactamente así**:
 - historialID
 - titulo
 - descripcion
 3. Programa el método **insertar de la clase ClienteDAO** teniendo en cuenta que debe devolver el clienteID generado en caso de que todo haya ido correctamente
 4. Programa el método **obtenerTodos de la clase ClienteDAO**
 5. Programa el método **obtenerPorId** de la clase **ClienteDAO**
 6. Programa el método **actualizar** de la clase **ClienteDAO**
 7. Programa el método **eliminar** de la clase **ClienteDAO**
 8. Programa en el **VideojuegoService** el **método insertar** teniendo en cuenta lo siguiente:
 - Debes hacer las **comprobaciones oportunas** para que no haya posibles errores descontrolados (salvo SQLExceptions que no podemos controlar)
 - Debes **insertar tanto el videojuego como la historia asociada** al mismo tiempo. **Controla la transaccionalidad**
 - **Gestiona todos los posibles errores** devolviendo cada código de error oportuno en cada caso
 9. Programa en el **VideojuegoService** el **método actualizar** teniendo en cuenta que es importante **reutilizar el código**

10. Programa en **PedidoService** el **método eliminar** teniendo en cuenta lo siguiente:
- Debes hacer las **comprobaciones oportunas** para que no haya posibles errores descontrolados (salvo SQLExceptions que no podemos controlar)
 - Debes **eliminar el pedido y sus “contiene” asociados** al mismo tiempo. **Controla la transaccionalidad**
 - **Gestiona todos los posibles errores** devolviendo cada código de error oportuno en cada caso
11. Programa en el **Main** el método **tuTest** haciendo las siguientes acciones:
- Crea un videojuego con su historia asociada
 - Crea un cliente
 - Crea un pedido con 2 juegos contenidos
 - Elimina todo lo que has creado evitando los errores

Rúbrica de corrección

Paso (pts)	Corrección	Pts.
1 (0.5)		
2 (0.5)		
3 (0.75)		
4 (0.75)		
5 (0.5)		
6 (0.5)		
7 (0.5)		
8 (2.5)		
9 (1)		
10 (2)		
11 (0.5)		