Innovación en Formación Profesional

Módulo Profesional 03:

Programación

Actividad Evaluable UF6

CICLO FORMATIVO DE GRADO SUPERIOR EN

DESARROLLO DE APLICACIONES

MODALIDAD ONLINE

Presentación de actividades

Objetivos

- Conocer los diferentes tipos de BBDD: relacional, relacional-objeto y orientada a objetos.
- Gestionar información almacenada en una BBDD, realizando las operaciones clásicas de inserción, modificación, borrado y consulta.
- Conectarnos a una BBDD de manera eficiente.
- Recorrido de objetos para obtener la información de la BBDD.
- Realizar el mapeo entre los datos de una BBDD y un objeto.
- Desarrolla programas aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.
- Escribe programas que manipulen información seleccionando y utilizando los tipos avanzados de datos facilitados por el lenguaje.
- Gestiona los errores que pueden aparecer en los programas, utilizando el control de excepciones facilitado por el lenguaje.

Competencias asociadas:

- Configurar y explotar sistemas informáticos, adaptando la configuración lógica del sistema según las necesidades de uso y los criterios establecidos

Metodología

- Preparación individual
- Para la realización del ejercicio se deberán visualizar todos los videotutoriales del curso, incluyendo los relativos a la solución de ejercicios propuestos.
- Videoconferencias.
- Píldoras informativas.

Entrega

2 de Mayo de 2023.

Se deberá entregar un único proyecto en Java comprimido en **Zip** con el siguiente formato:

MP03_UF06_Apellido_Nombre.zip

Por ejemplo:

MP03_UF06_Perez_Juan.zip

Dedicación estimada

8 horas

Documentos de referencia

Videotutoriales de la UF.

Libro de referencia de la asignatura.

Videotutoriales de ejercicios resueltos.

Resultados de aprendizaje

- Gestiona información almacenada en bases de datos relacionales manteniendo la integridad y consistencia de los datos.
- Gestiona información almacenada en bases de datos objeto- relacionales manteniendo la integridad y consistencia de los datos

- Utiliza bases de datos orientadas a objetos, analizando sus características y aplicando técnicas para mantener la persistencia de la información
- Reconocer y aplicar los fundamentos de la POO.
- Desarrollar programas organizados en clases.
- Desarrolla programas aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.
- Escribe programas que manipulen información seleccionando y utilizando los tipos avanzados de datos facilitados por el lenguaje.
- Gestiona los errores que pueden aparecer en los programas, utilizando el control de excepciones facilitado por el lenguaje.

Criterios de evaluación

- Realizar las operaciones básicas de gestión de información con una BBDD: Acceso, consulta, modificación, inserción y borrado de datos.
- Aplicar los conceptos fundamentales de la POO: composición, herencia y polimorfismo (sobrecarga de métodos), sobreescritura, constructores, atributos y métodos de una clase, etc.
- Organización del código en clases y paquetes.
- Uso de estructuras de datos avanzadas.
- Uso de librerías adecuadas.
- Gestión de errores en el código.
- Claridad y eficiencia en el código.
- Modularidad adecuado en el código.
- Documentación del código.

Descripción de la actividad

- Este reto evaluable te servirá para aprender a afianzar todos los conocimientos adquiridos durante la UF6. Practicarás el mecanismo de acceso a una BBDD, el mapeo entre los datos de una BBDD y objetos, continuaremos practicando el uso de estructuras de datos avanzadas y otros contenidos vistos en unidades formativas anteriores.
- **Se deberá entregar con el formato correcto, tal y como se indica en este documento. En caso contrario se descontará 2 punto.**
- **El ejercicio deberá poder ejecutarse sin problema, es decir, no deben existir errores en la compilación (en caso de tener errores el código el ejercicio evaluable será suspenso).**
- **La entrega tardía del ejercicio restará 1,5 punto. Dicha entrega podrá realizarse hasta un día después de la fecha de vencimiento del ejercicio que se indicará en la sección de entrega del ejercicio.**
- **Previa a la fecha de entrega de este ejercicio, no se revisará código alguno por parte del profesor.**
- Nota: Es mejor entregar pocos apartados que estén bien, que muchos y no compile.
- Nota II: Prueba, prueba y prueba.

Desarrollo de la actividad

El ejercicio evaluable consistirá en realizar un programa aplicando lo visto en la UF6, gestionando información sobre una BBDD postgreSQL.

Ejercicio 1:

(10 puntos)

Acceso y operaciones sobre una BBDD PostgreSQL.

Dado el script con la creación de la tabla empleados adjunta al ejercicio, se desea implementar una gestión que realice las siguientes operaciones sobre la BBDD por este orden:

- Crea una clase que tenga como atributos los datos de la tabla empleados. Con sus constructores, y los métodos correspondientes.
- En el programa principal, crea varios objetos de tipo empleado.
- Crea una clase que gestione las operaciones de BBDD, implementando los siguientes métodos:
    - Conectar a la bbdd.
    - Cerrar los recursos de la BBDD.
    - Insertar un objeto de tipo empleado.
    - Borra un registro a partir de un objeto de tipo empleado.
    - Recupera todos los registros de la BBDD. Este método deberá retornar una estructura de datos compleja, ya sea unidimensional o bidimensional.

Se adjuntan las cabeceras de los métodos al final de este documento. También se adjunta una clase con las consultas SQL a realizar.

- En el programa principal, deberás crear los objetos de tipo empleado, crear una instancia de la clase de utilidad para las operaciones de BBDD y realizar las siguientes operaciones en este orden:
    - Muestra todos los registros de la tabla.
    - Inserta varios registros en la tabla. Los datos de entrada se simularán en el programa principal, bien a través de la consola, bien directamente en el código.
    - Muestra todos los registros de la tabla.
    - Elimina uno de los registros.
    - Muestra todos los registros de la tabla.

Otros requisitos relativos al comportamiento de la aplicación:

- Todos los datos serán necesarios en el momento de la creación de los objetos correspondientes y se permitirá la modificación de todos los datos exceptuando el identificador del empleado que no se permitirá su cambio.
- Gestiona los errores utilizando excepciones, evitando la llamada a la instrucción e.printStackTrace();

Otros requisitos de codificación:

- La base de datos se llamará empresadb.
- Modulariza en la medida de lo posible.
- Se adjunta clase con las consultas SQL en la tarea.
- La solución deberá estar en un proyecto exclusivo para este ejercicio.
- Formatea adecuadamente los datos que se muestran en consola.
- El código de este ejercicio deberá incluirse en el paquete *es.ifp.programacion.ejercicio.uf6*, pudiendo haber más paquetes bajo esta ruta.
- Adjunta la documentación en formato *Javadoc*. Se permitirá que no haya acentos para que no haya problemas de codificación en el HTML resultante.

Criterios de calificación:

- Estructura de clases (2 puntos).
- Buenas prácticas en el código y gestión de errores (2 puntos).
- Aplicación de las características de la POO y acceso a BBDD(2 puntos).
- Eficiencia (1 punto)
- Explicación de la solución implementada y justificación (0,5).
- Documentación del código en sus partes más complejas (0,5).

Cabecera de las funciones a implementar en la clase de utilidades de BBDD

```
/**
 * Borra el registro de la tabla asociado al objeto emp pasado como parámetro
 * @param emp objeto Empleado que se elimina de la BBDD
 * @return true si el borrado fue correcto, false sino.
 */
public boolean deleteEmpleado(Empleado emp);

/**
 * A partir de un objeto empleado, inserta todos sus datos en la tabla empleados.
 * @param emp Objeto de tipo empleado
 * @return 0 si no se ha insertado valor alguno. En caso contrario, el número de filas insertadas.
 */
public int insertarEmpleado(Empleado emp);


/**
 * Cierra todos los recursos relativos a la BBDD
 * @return true si la operación fue correcta, false sino.
 */
public boolean closeResources();


/**
 * Se conecta a la BBDD empresadb creada en postgreSQL.
 * @return true si la conexión fue satisfactoria, false sino.
 */
public boolean connectToDB();


/**
 * Obtiene todos los registros de la tabla empleados
 * @return un ArrayList de objetos de tipo Empleado mapeados a los registros de la tabla.
```

```
 */
public ArrayList<Empleado> getAllEmpleados();
```