

PRESA. UNIDAD 2. ACCESO A BASES DE DATOS RELACIONALES PARTE 1. EJERCICIOS NO EVALUABLES

DAM. Acceso a Datos (ADA) (a
distancia en inglés)

Unidad 2. ACCESO A BASES DE DATOS RELACIONALES

Parte 1. Trabajar con Bases de Datos Relacionales. Ejercicios no
evaluables

Abelardo Martínez

Basado y modificado de Sergio Badal (www.sergiobadal.com)

Año 2023-2024

Aspectos a tener en cuenta

Importante

Si buscas las soluciones navegando por Internet o preguntando al oráculo de ChatGPT te estarás engañando. Tenga en cuenta que ChatGPT no es infalible ni todopoderoso.

Es una gran herramienta para acelerar tu trabajo una vez que dominas una materia, pero utilizarla como atajo a la hora de adquirir habilidades y conocimientos básicos socava gravemente tu aprendizaje. Si lo utiliza para obtener soluciones o consejos por su cuenta, consulte también detenidamente las soluciones propuestas. Intenta resolver las actividades utilizando los recursos que hemos visto y la documentación ampliada que encontrarás en el "Aula Virtual".

Consejos para programar

Recomendamos seguir los siguientes estándares de codificación:

- Una instrucción por línea.
 - Agregue comentarios para que su código sea más claro y legible.
 - Utilizar la notación húngara para reconocer el tipo de variables a primera vista.
 - Recuerda que hay varias formas de implementar una solución, así que elige la que más te guste.
- Recomendamos encarecidamente utilizar soluciones basadas en buffer.

1. Modo consola. Administrar una base de datos relacional a través de JDBC en SQLite

Actividad (no evaluable)

Crear un programa en Java para gestionar PIZZAS en un restaurante italiano imprimiendo y utilizando un menú específico. Después de cada opción, el usuario debería ver el mismo menú hasta que se presione la opción cero. No dudes en compartir tus dudas en el foro UNIT. Puedes duplicar este proyecto de la UNIDAD 1.

ATENCIÓN: Utilice las excepciones adecuadas al acceder a bases de datos.



Opciones de menú:

- Presione 0 para "Salir"
- Presione 1 para "Pedir pizzas hasta que el usuario ingrese cero como ID"
 - Para cada pizza necesitamos el ID (Entero), el Nombre (Cadena con espacios) y el Precio (Doble), agregados al ArrayList de pizzas.
 - Una vez que se ingresa cero como ID, todas las pizzas se guardarán en una base de datos SQLite usando JDBC. Tú decides los campos y el nombre de la tabla.
 - ATENCIÓN: ¡El ID debe ser un número entero! Para cada pizza, debes pedir un número entero (en bucle) hasta que el usuario ingrese un número entero o cero para imprimir nuevamente el menú.
 - ATENCIÓN: La tabla debe descartarse y crearse nuevamente cada vez que agregue la Lista de arreglo.
- Presione 2 para "Enumerar todas las pizzas"
 - Simplemente lee la base de datos SQLite usando JDBC e imprima la información de cada pizza.
- Presione 3 para "Eliminar todas las pizzas"
 - Simplemente elimine las pizzas almacenadas en la base de datos SQLite usando JDBC.

Ejemplo de menú:

MENÚ

=====

- 0. Salir
- 1. Agrega pizzas
- 2. Enumera todas las pizzas
- 3. Retire todas las pizzas

=====

Seleccione una opción:

2. Modo consola. Administrar una base de datos relacional a través de JDBC en MySQL

Actividad (no evaluable)

Crear un programa en Java para gestionar ANIMALES en una clínica veterinaria imprimiendo y utilizando un menú específico. Después de cada opción, el usuario debería ver el mismo menú hasta que se presione la opción cero. No dudes en compartir tus dudas en el foro UNIT. Puedes duplicar este proyecto de la UNIDAD 1.

ATENCIÓN: Utilice las excepciones adecuadas al acceder a bases de datos.



Opciones de menú:

- Presione 0 para "Salir"
- Presione 1 para "Preguntar por animales (pacientes) hasta que el usuario ingrese cero como ID"
 - Para cada animal necesitamos el ID (Cadena), Nombre (Cadena con espacios), Especie (Cadena), Raza (Cadena con espacios) y Edad actual (Entero) y Esterilizado (Cadena), guardados en un ArrayList de animales. Si un animal está esterilizado, escribimos "sí" y "no" en caso contrario.
 - Una vez que se ingresa cero como ID, todos los animales se guardarán en una base de datos MySQL usando JDBC. Tú decides los campos y el nombre de la tabla.
 - ATENCIÓN: ¡La edad debe ser entera!
 - ATENCIÓN: La tabla debe descartarse y crearse nuevamente cada vez que agregue la
Lista de arreglo.
- Presione 2 para "Enumerar todos los animales"
 - Simplemente lea la base de datos MySQL usando JDBC e imprima la información de cada animal.
- Presione 3 para "Eliminar todos los animales"
 - Simplemente elimine los animales almacenados en la base de datos MySQL usando JDBC.

Ejemplo de menú:

```
*****  
  
MENÚ  
*****  
  
=====
```

0. Salir
1. Agrega animales
2. Enumere todos los animales
3. Retire todos los animales.

```
=====
```

Seleccione una opción:



Licenciado bajo la [licencia Creative Commons Attribution Share Alike 4.0](https://creativecommons.org/licenses/by-sa/4.0/)