

ADMINISTRACIÓN DE BASE DE DATOS

Nombre del alumno: Isaac Brandon Martínez Ramírez

Matricula: 22SIC008 Fecha: 18/Julio/2024 **3° Parcial 6° A - B** Mayo – Agosto 2024 **Ingeniería en Sistemas Computacionales**

1. Es un script que se usa en lenguaje de programación SQL: •

Trigger o Dispador

2. Se aplican a la base de datos cuando se realizan operaciones en la tabla, añadir, actualizar o eliminar:

Trigger

3. Contribuye a mejorar la gestión de la base de datos:

Es el índice

4. Aquellos cuya ejecución se realiza a través de llamadas desde una tabla asociada:

Procedimientos Almacenados

5. Se ejecutan solo una vez, independientemente de la cantidad de veces que se cumplan las condiciones para su ejecución:

Trigger After

6. Consiste en programar el envío de notificaciones o mensajes personalizados a los clientes:

Procedimientos Almacenado

7. Operación que devuelve a la base de datos a algún estado previo:

Rollback

8. Operación que confirma un conjunto de cambios provisionales de forma permanente.

Commit

9. Puntos concretos en la transacción donde poder deshacer la transacción hasta esos puntos

Savepoint

10. Es una serie de una o más operaciones ejecutadas como una única unidad atómica de trabajo:

Trasaction

20. Relaciona la acción con el comando:

- Ejecutar sentencia :

i. Comando SQL (SELECT, INSERT, UPDATE, DELETE, etc.)

- Después de ejecutar la sentencia y comprobar las restricciones y condiciones aplicables :

i. AFTER Trigger

- Antes de modificar la fila de la tabla afectada por la sentencia del trigger, y de comprobar las restricciones y condiciones de ejecución :

i. BEFORE Trigger

- Después de modificar la fila de la tabla afectada por la sentencia del trigger y de comprobar las restricciones y condiciones de ejecución:

i. AFTER Trigger

11. Garantizan la coherencia e integridad de los datos, asegurando que la base de datos siga siendo coherente incluso en caso de fallos o errores del sistema:

Transacciones ACID (Atomicidad,

Consistencia, Aislamiento, Durabilidad)

12. Proceso mediante el que se migran datos de una o más bases de datos de origen a una o más bases de datos de destino mediante un servicio de migración de bases de datos.

ETL o Proceso de Migración de Datos

13. Base de datos que contiene datos para migrar a una o más bases de datos de destino:

Base de Datos Origen

14. Base de datos que recibe datos migrados desde una o más bases de datos de origen:

Base de Datos de Migración

15. Migración desde las bases de datos de origen hacia las bases de datos de destino en la que las bases de origen y de destino pertenecen al mismo sistema de administración de bases de datos:

Migración Homogénea

16. Migración desde las bases de datos de origen hacia las bases de datos de destino en la que las bases de datos de origen y de destino pertenecen a diferentes sistemas de administración de bases de datos,

Migración Heterogénea

17. Transfiere datos de forma continua desde las bases de datos de origen hacia las bases de datos de destino, sin borrar las de origen.

Replicación de Datos

18. Usuarios que han abierto un email sobre el total de los enviados:

Tasa de Apertura

19. Usuarios que han clickeado en un email sobre el total de los enviados:

Tasa de Clics

ADMINISTRACIÓN DE BASE DE DATOS

Examen práctico

Nombres del alumno: Isaac Brandon Martínez Ramírez

Matricula: 22SIC008 Fecha: 18/Julio/2024 **3° Parcial 6° A - B** Mayo – Agosto 2024 **Ingeniería en Sistemas Computacionales**

1. Haz que no se pueda añadir un nuevo animal si el tipo es 'León' y el número de años es mayor que 20.
2. Cuando se añada un nuevo animal, hacer que dicho animal esté cuidado por el artista que cuida a menos animales.
3. Haz todo lo necesario para que el campo ganancias de la tabla ATRACCIONES se actualice cuando se añadan, borren o modifiquen datos en la tabla ATRACCION_DIA.
4. Modifica la tabla ATRACCIONES y añade una nueva columna de nombre contador, de tipo numérico y valor por defecto de cero, que lleve cuenta de cuantas veces se ha celebrado la atracción y que se actualice con cualquier operación. Crea los triggers necesarios.
5. Impide que se pueda añadir o modificar una pista con un aforo superior a 1000 o inferior a 10 (esto lo podríamos implementar con un check, pero vamos a practicar el uso de triggers).
6. Haz que si se intenta dar de alta un nuevo artista y se envía un nif_jefe que no exista, se cambie su valor por null.
7. Crea una tabla de nombre REGISTRO con las columnas:
 - id autonumérica Clave primaria
 - usuario: varchar(100)
 - tabla: varchar(100)
 - operacion: varchar(10)
 - datos_antiguos: varchar(100) (guardarán los datos nombre_pista:aforo que se borren o modifiquen)
 - datos_nuevos: varchar(100) (guardarán los datos nombre_pista:aforo que se añadan o modifiquen)
 - fecha-hora: datetime

Haz que se registren las operaciones de alta, baja y modificación sobre la tabla 'PISTAS'.

a. Por ejemplo:

 - 'angel' - 'PISTAS' - 'ALTA' - null - 'pista_nueva:1000' - '01-01-2000 17:00:00' (el campo datos_antiguos es nulo ya que estamos a dar de alta una nueva pista).
 - 'luis' - 'PISTAS' - 'BAJA' - 'pista_borrar:1000' - null - '01-01-2000 18:00:00' (el campo datos_nuevos es nulo ya que estamos a dar de baja una pista).
 - 'pepe' - 'PISTAS' - 'MODIFICAR' - 'pista_modificar:1000' - 'pista_modificar:1500' - '01-01-2000 19:00:00' (el campo datos_nuevos guarda los datos modificados y el campo datos_antiguos guarda los datos antes de la modificación).
 - Si ya existen triggers creados, nos los modifiques, crea nuevos.
8. Crea una tabla de nombre CONTADOR con las columnas:
 - a. id autonumérica Clave primaria
 - b. tipo: varchar(100) no nulo
 - c. valor: int no nulo
 - d. Añade dos filas con los valores para tipo/valor: pistas/0 animales/0
 - e. Haz que cada vez que haya alguna operación que modifique (alta/baja) el número de pistas o de animales, se actualice el número total de los mismos.
 - f. Ejecuta la orden SQL que actualice la tabla contador con los datos actuales de las tablas.
9. Ejecuta la orden SQL que muestre los triggers creados en la tabla PISTAS.
10. Ejecuta la orden SQL que muestre la información del trigger del ejercicio 1 (de nombre animales_checkAdd_INSERT).

New Database

Select a page

General

Options

Filegroups

Connection

Server:
PC-BRAN\SQLEXPRESS

Connection:
PC-BRAN\isabr

View connection properties

Progress

Ready

Script Help

Database name:

zoologico

Owner:

<default>

...

Use full-text indexing

Database files:

Logical Name	File Type	Filegroup	Initial Size (MB)	Autogrowth / Maxsize	Path
zoologico	ROWS ...	PRIMARY	8	By 64 MB, Unlimited	C:\Progra
zoologico_log	LOG	Not Applicable	8	By 64 MB, Unlimited	C:\Progra

Add

Remove

OK

Cancel

```
        tabla VARCHAR(100),
        operacion VARCHAR(10),
        datos_antiguos VARCHAR(100),
        datos_nuevos VARCHAR(100),
        fecha_hora DATETIME
    );

-- Tabla de contador
CREATE TABLE CONTADOR (
    id INT PRIMARY KEY IDENTITY(1,1),
    tipo VARCHAR(100) NOT NULL,
    valor INT NOT NULL
);

-- Insertar valores iniciales en CONTADOR
INSERT INTO CONTADOR (tipo, valor) VALUES ('pistas', 0), ('animales', 0);
GO
```

100 %

Messages

(2 rows affected)

Completion time: 2024-07-19T21:58:49.0624555-06:00