**GUÍON DE ACTIVIDAD EVALUABLE**

* **Ciclo**: Administración de Sistemas Informáticos en Red (ASIR)
* **Asignatura**: Gestión de Bases de Datos (GBD)
* **Curso**: Primero

**ACTIVIDAD EVALUABLE 5 (AEV – 5)**

***TÍTULO DE LA ACTIVIDAD:***

Gestión de una base de datos.

***OBJETIVOS:***

Esta actividad tiene como objetivo reforzar lo estudiado sobre SQL.

***TEMPORALIZACIÓN:***

La duración de esta actividad será de 4 horas máximo.

***PROCESO DE DESARROLLO:***

El alumno deberá:

1. Consultar apuntes, videos, webinars y tareas realizadas.
2. Resolver las consultas propuestas.

***ENTREGABLES:***

**Se entregará:**

* Un documento WORD, en dicho documento para cada consulta SQL en modo texto (que se pueda copiar y pegar para ejecutarlas) y pantallazos de la ejecución de las diferentes consultas.
* Base de datos exportada en -sql una vez finalizas todas las consultas.

***ENUNCIADO:***

A partir de la Bases de datos adjunta (AEV5BD.txt) realiza las siguientes consultas:

1. Crea una tabla llamada GESTIÓN que contenga los siguientes campos:

|  |  |
| --- | --- |
| **TABLA GESTIÓN** | |
| **Campo** | **Tipos datos** |
| **Cod\_Registro** | Int (4) / Auto\_increment / Primary key |
| **Cod\_Tratamiento** | Int(4) |
| **Descripcion\_Ant** | Varchar (25) |
| **Descripcion\_Nuevo** | Varchar (25) |
| **Fecha\_Registro** | Datetime |
| **Precio\_Ant** | Decimal (5,2) |
| **Precio\_Nuev** | Decimal (5,2) |
| **Cantidad\_Ant** | Int(11) |
| **Cantidad\_Nueva** | Int(11) |
| **Acción** | Varchar (20) |
| **Usuario** | Varchar (20) |
|  |  |

1. Cada vez que se inserte un nuevo tratamiento en la tabla TRATAMIENTOS, se tendrá que insertar de manera automática en la tabla GESTION, una fila indicando el código del tratamiento, la descripción del tratamiento, la fecha de registro de la acción, el precio del tratamiento, la cantidad que hay de tratamientos, la acción que se ha realizado (en este caso ‘NUEVO’) y el usuario que ha registrado la consulta.

**CREATE TRIGGER nuevo\_tratamiento**

**AFTER INSERT ON tratamientos**

**FOR EACH ROW**

**INSERT INTO gestion(Cod\_Tratamiento, Descripcion\_nuevo, Fecha\_registro, Precio\_nuevo, Cantidad\_nueva, Accion, Usuario)**

**VALUES (NEW.Cod\_tratamiento, NEW.Descripcion, NOW(), NEW.Precio, NEW.Cantidad, 'NUEVO', CURRENT\_USER())**

1. Cada vez que se modifique un tratamiento en la tabla TRATAMIENTOS, se tendrá que insertar de manera automática en la tabla GESTION, una fila indicando el código de la tratamiento, la descripción del tratamiento anterior y la nueva, la fecha de registro de la acción, el precio del tratamiento anterior y el nuevo, la cantidad de tratamientos anterior y nueva, la acción que se ha realizado (en este caso ‘MODIFICACIÓN) y el usuario que ha registrado la consulta.

**CREATE TRIGGER modificar\_tratamiento**

**BEFORE UPDATE ON tratamientos**

**FOR EACH ROW**

**INSERT INTO gestion(Cod\_Tratamiento, Descripcion\_ant, Descripcion\_nuevo, Fecha\_registro, Precio\_ant, Precio\_nuevo, Cantidad\_ant, Cantidad\_nueva, Accion, Usuario)**

**VALUES (OLD.Cod\_tratamiento, OLD.Descripcion, NEW.Descripcion, NOW(), OLD.Precio, NEW.Precio, OLD.Cantidad, NEW.Cantidad, 'MODIFICACION', CURRENT\_USER())**

1. Cada vez que se borre un tratamiento en la tabla TRATAMIENTOS, se tendrá que insertar de manera automática en la tabla GESTION, una fila indicando el código del tratamiento, la descripción del tratamiento, la fecha de registro de la acción, el precio del tratamiento, la cantidad que hay de tratamientos, la acción que se ha realizado (en este caso ‘ELIMINACIÓN) y el usuario que ha registrado la consulta.

**CREATE TRIGGER eliminar\_tratamiento**

**AFTER DELETE ON tratamientos**

**FOR EACH ROW**

**INSERT INTO gestion(Cod\_Tratamiento, Descripcion\_ant, Fecha\_registro, Precio\_ant, Cantidad\_ant, Accion, Usuario)**

**VALUES (OLD.Cod\_tratamiento, OLD.Descripcion, NOW(), OLD.Precio, OLD.Cantidad, 'ELIMINACION', CURRENT\_USER())**

1. Cada vez que se añade una consulta de algún tratamiento, se tiene que actualizar la cantidad de tratamientos que hay. (Cada consulta es un tratamiento menos).

**CREATE TRIGGER añadir\_consulta**

**AFTER INSERT ON consultas**

**FOR EACH ROW**

**UPDATE tratamientos**

**SET Cantidad = Cantidad - 1**

**WHERE Cod\_tratamiento = NEW.Cod\_tratamiento;**

1. Crea un procedimiento almacenado para consultar los clientes que tienen menos de 30 años.
2. Crea un procedimiento almacenado de las mascotas que han realizado alguna consulta antes de enero de 2021 (no incluido)
3. Crea un procedimiento almacenado para consultar los clientes que tienen mascotas de tipo ‘perro’.
4. Crea un procedimiento almacenado que, indicando el código del tratamiento y el precio nuevo, se actualice el precio del tratamiento.
5. Crea un procedimiento almacenado para calcular el precio con descuento del 10% del tratamiento que indique el usuario. Hazlo definiendo variables (descuento=0.10). Se debe mostrar el precio antiguo y el nuevo.
6. Crea un Trigger para que cada vez que se modifique el precio de los tratamientos, si se quiere poner un número menor de 20 que te ponga un 20 en el precio, si el precio es mayor que 500 que ponga un 500 y si está entre 20 y 500 que ponga el precio que se indica el usuario.

**CREATE TRIGGER modificar\_precio**

**BEFORE UPDATE ON tratamientos**

**FOR EACH ROW**

**BEGIN**

**IF NEW.Precio < 20 THEN**

**SET NEW.Precio = 20;**

**ELSEIF NEW.Precio > 500 THEN**

**SET NEW.Precio = 500;**

**END IF;**

**END;**

***EVALUACIÓN:***

* Esta actividad es puntuable y, por tanto, OBLIGATORIA su entrega.
* La evaluación de esta será siguiendo la siguiente baremación:

|  |  |
| --- | --- |
| **INSUFICIENTE** | Falta más de 5 consultas o no se entrega alguna de la documentación requerida. |
| **SUFICIENTE** | Faltan menos de 5 consultas y las demás se realizan de forma adecuada. |
| **BIEN** | Faltan menos de 3 consultas y las demás se realizan de forma adecuada. |
| **NOTABLE** | Faltan menos de 2 consultas y las demás se realizan de forma adecuada. |
| **EXCELENTE** | Todas las consultas funcionan de forma adecuada. |