

**Asignación**

Modelo Relacional Proyecto 1

Fecha

02/09/2024

Materia

Bases de Datos Avanzadas

Profesor

Alma Leticia Chavez Quintero

Procedures y triggers para funciones genéricas de la base de datos, en el script esta todo comentado

DELIMITER \$\$

```
CREATE PROCEDURE passw(IN idtransaccion INT)
BEGIN
    DECLARE caracteres VARCHAR(100) DEFAULT
'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789'
    DECLARE password VARCHAR(255) DEFAULT "";
    DECLARE i INT DEFAULT 0;
    WHILE i < 8 DO
        SET password := CONCAT(password, SUBSTRING(caracteres,
        FLOOR(1 + RAND() * LENGTH(caracteres)), 1));
        SET i := i + 1;
    END WHILE;
    UPDATE TransaccionFolioCliente SET password_transaccion = password
    WHERE id_transaccion = idtransaccion;

    END $$
DELIMITER ;
```

DELIMITER \$\$

CREATE PROCEDURE folio(IN idtransaccion INT)

BEGIN

DECLARE caracteresFolio VARCHAR(100) DEFAULT
'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789'

DECLARE folio VARCHAR(255) DEFAULT '';

DECLARE i INT DEFAULT 0;

WHILE i < 20 DO

SET folio := CONCAT(folio, SUBSTRING(caracteresFolio, FLOOR(1 +
RAND() * LENGTH(caracteresFolio)), 1));

SET i := i + 1;

END WHILE;

UPDATE TransaccionFolioCliente SET folio_transaccion = folio WHERE
id_transaccion = idtransaccion;

END\$\$

DELIMITER;

DELIMITER \$\$

CREATE TRIGGER tipo_transaccion

AFTER INSERT ON transaccion

FOR EACH ROW

BEGIN

 DECLARE id int;

 SET id = NEW.id_transaccion;

 IF NEW.tipo_transaccion = false THEN

 INSERT INTO TransaccionFolioCliente (id_transaccion,estado) VALUES
(NEW.id_transaccion, 0);

 CALL folio(id);

 CALL passw(id);

 END IF;

END\$\$

DELIMITER ;

DELIMITER \$\$

CREATE PROCEDURE verHistorial(IN id INT)

BEGIN

 SELECT

 id_transaccion AS 'Id',

 fecha_transaccion AS 'Fecha',

 hora_transaccion AS 'Hora',

 cantidad AS 'Monto',

 CASE

 WHEN tipo_transaccion = 0 THEN 'Folio'

 WHEN tipo_transaccion = 1 THEN 'Cliente'

 END AS 'Tipo de Transacción'

FROM

 transaccion

 WHERE id = transaccion.id_cuenta;

END\$\$

DELIMITER ;

DELIMITER \$\$

CREATE TRIGGER calcularEdad

BEFORE INSERT ON cliente

FOR EACH ROW

BEGIN

SET NEW.edad =

TIMESTAMPDIFF(YEAR,NEW.fecha_nacimiento,CURDATE());

END\$\$

DELIMITER ;

DELIMITER \$\$

CREATE PROCEDURE transferencia(IN id_cuenta_origen INT, monto INT,
id_cuenta_destino INT)

BEGIN

DECLARE saldo_origen INT;

SELECT saldo INTO saldo_origen FROM cuenta WHERE id_cuenta =
id_cuenta_origen;

IF saldo_origen >= monto THEN

START TRANSACTION;

UPDATE cuenta SET saldo = saldo - monto WHERE id_cuenta =
id_cuenta_origen;

UPDATE cuenta SET saldo = saldo + monto WHERE id_cuenta =
id_cuenta_destino;

COMMIT;

else

ROLLBACK;

END IF;

END\$\$

DELIMITER ;

TABLAS

Cliente (

id INT PRIMARY KEY AUTO INCREMENT,
nombre VARCHAR(50) NOT NULL,
apellido_paterno VARCHAR(50) NOT NULL,
apellido_materno VARCHAR(50) NOT NULL,
calle VARCHAR(255) NOT NULL,
colonia VARCHAR,
codigo_postal VARCHAR(255) NOT NULL,
fecha_nacimiento DATE NOT NULL,
edad INT date() - fecha_nacimiento,

);

Cuenta (

id_cuenta INT PRIMARY KEY NOT NULL //numero de cuenta
fecha_apertura DATE NOT NULL
saldo BIGINT
id_cliente INT,
FOREIGN KEY (id_cliente) **REFERENCES** Cliente(id)

);

Transaccion (

id_transaccion INT PRIMARY KEY NOT NULL,
fecha_transaccion DATE NULL DEFAULT NULL,
hora_transaccion TIME NULL DEFAULT NULL,
cantidad INT NOT NULL
tipo_transaccion BOOLEAN DEFAULT TRUE

id_cuenta INT

FOREIGN KEY (id_cuenta) **REFERENCES** Cuenta(id_cuenta)

);

TransaccionNormalCliente (

id_transaccion INT, // guardamos sin pk, solo referenciamos a la entidad padre

FOREIGN KEY id_transaccion **REFERENCES**

Transaccion(id_transaccion)

);

TransaccionFolioCliente (

id_transaccion INT,

folio_transaccion VARCHAR(255),

password_transaccion VARCHAR(8),

estado boolean DEFAULT FALSE,

FOREIGN KEY id_transaccion **REFERENCES**

Transaccion(id_transaccion)

);