UNIVERSIDAD PRIVADA "FRANZ TAMAYO"



EVALUACION HITO 4



BASE DE DATOS II

ALUMNO: Adán Abinadi Tudela Cabero

Evaluacion - Ing. De Sistemas

DOCENTE: Lic. William R. Barra Paredes.

Cochabamba, 9 de diciembre de 2019

${f 1.}$ Resolver los siguientes ejercicios.

- Generar la serie fibonacci.
 - El objetivo es sumar todos los números de la serie fibonacci desde una cadena.
 - Es decir usted tendrá solo la cadena generado con los primeros N números de la serie fibonacci y a partir de ellos deberá sumar los números de esa serie.
 - Puede utilizar el siguiente código para generar la serie fibonacci.

```
CREATE FUNCTION sumaFibonacci(limite INTEGER) RETURNS TEXT
    DECLARE respuesta TEXT DEFAULT '0';
    DECLARE x INTEGER DEFAULT 0;
    DECLARE y INTEGER DEFAULT 1;
    DECLARE z INTEGER DEFAULT 0;
    DECLARE i INTEGER DEFAULT 0;
    DECLARE sum BIGINT DEFAULT 0;
    REPEAT
        SET z = x + y;
        SET x = y;
        SET y = z;
        SET sum = sum + x;
        SET respuesta = CONCAT(respuesta,', ',x);
        SET i = i + 1;
    UNTIL i > (limite-2) END REPEAT ;
    RETURN CONCAT(respuesta,' - SUMATOTAL: ',sum);
END;
SELECT sumaFibonacci(10);
```

- Crear una Vista de nombre detalle_proyecto_persona de las personas que no hayan nacido en la ciudad de COCHABAMBA.
 - La Vista debe de tener los siguientes campos (NOMBRES, APELLIDOS, PROYECTO, TIPO_PROYECTO, NIVEL_PROYECTO, SEXO y DEPARTAMENTO).
 - Consideraciones de la columna NIVEL_PROYECTO
 - Si la edad es mayor que 0 y menor igual a 20 mostrar 'MI PRIMER PROYECTO'
 - Si la edad es mayor que 20 y menor igual a 25 mostrar 'MI SEGUNDO PROYECTO'
 - Si la edad es mayor que 25 y menor igual a 30 mostrar 'MI TERCER PROYECTO'
 - Considere la imagen siguiente para tener el contexto de qué datos debe de mostrar cada columna.

```
• CREATE VIEW detalle_proyecto_persona AS

SELECT P.nombre NOMBRES,

P.apellidos APELLIDOS,

PRY.nombreProy PROYECTO,

PRY.tipoProy TIPO_PROYECTO,

case
```

```
when P.edad <=20 then 'MI PRIMER PROYECTO'
    when P.edad <=25 then 'MI SEGUNDO PROYECTO'
    when P.edad <=30 then 'MI TERCER PROYECTO'
    ELSE 'CONTENIDO NO ENCONTRADO'
    end AS NIVEL_PROYECTO,
    P.sexo SEXO,D.nombre DEPARTAMENTO

FROM PERSONA AS P
    INNER JOIN DETALLE_PROYECTO DP on P.id_per = DP.id_per
    INNER JOIN DEPARTAMENTO D on P.id_dep = D.id_dep
    INNER JOIN PROYECTO PRY on DP.id_proy = PRY.id_proy
    WHERE D.nombre !='Cochabamba';
DROP VIEW detalle_proyecto_persona;</pre>
```

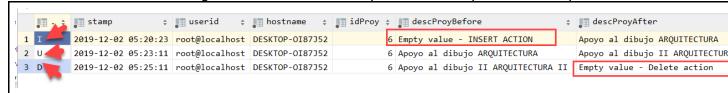
							•	
	NOMBRES \$	APELLIDOS \$	PROYECTO \$	TIPO_PROYECTO \$	NIVEL_PROYECTO \$	SEXO \$	DEPARTAMENTO \$	
1	nombre4	apellidos4	Alfabetizacion	EDUCACION	MI SEGUNDO PROYECTO	m	La Paz	
2	nombre4	apellidos4	Creacion de Escuelas	EDUCACION	MI SEGUNDO PROYECTO	m	La Paz	
3	nombre5	apellidos5	Apoyo al dibujo	ARQUITECTURA	MI TERCER PROYECTO	m	Santa Cruz	
4	nombre6	apellidos6	Apoyo al dibujo	ARQUITECTURA	MI PRIMER PROYECTO	f.	Santa Cruz	

 Crear TRIGGERS Before o After para INSERT, UPDATE y DELETE aplicado a la tabla PROYECTO.

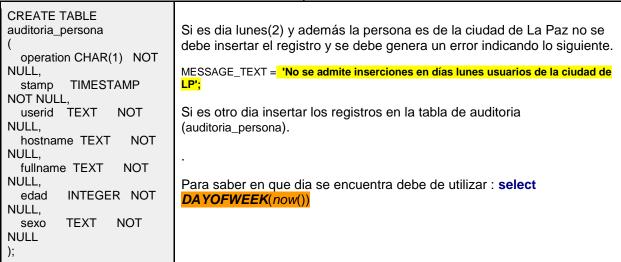
• Crear una tabla de auditoría de acuerdo al siguiente:

```
CREATE TABLE auditoria_proyecto
                                    Consideraciones:
  operation
             CHAR(1) NOT
                                    Los campos:
NULL.
                                    descProyBefore : Son los valores antes de la acción (insert -
             TIMESTAMP NOT
  stamp
                                    update - delete).
NULL,
                                    descProyAfter: Son los valores después de la acción (insert
  userid
            TEXT
                     NOT NULL.
                                    - update - delete).
              TEXT
  hostname
                       NOT NULL,
            INTEGER NOT
  idProy
                                    Nota. Debe crear triggers.
NULL,
  descProyBefore TEXT
                         NOT
  descProyAfter TEXT
                        NOT
NULL
);
```

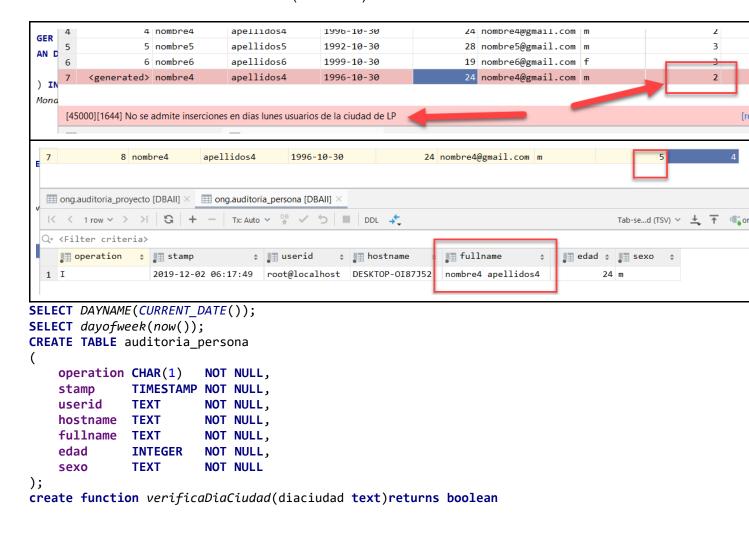
- Crear un procedimiento almacenado SP para insertar datos a la tabla de auditoría.
- Considere la siguiente tabla de auditoría para ver el comportamiento esperado.



Crear un TRIGGER BEFORE INSERT para la tabla PERSONA.



- Debe de crear una función para verificar que es dia lunes y que el usuario es de la ciudad de La Paz. la función recibe dos parámetros.
 - Dia de la semana(INTEGER).
 - o El ID de la ciudad (INTEGER)



```
begin
        declare dia integer default 0;
        declare ciuda integer default 0;
        declare res boolean default false;
        set dia = Locate('Wednesday',diaciudad);
        SET ciuda = locate('2',diaciudad);#laPaz es numero 2 en id
        if dia > 0 and ciuda >0
             then
             set res = true;
        end if;
        return res;
        #return if(diaciudad = 'Wednesday2',true,false);
    end:
drop function verificaDiaCiudad;
CREATE TRIGGER triggerPersonaver
     BEFORE INSERT ON PERSONA
   FOR EACH ROW
BEGIN
    DECLARE DIA boolean DEFAULT
verificaDiaCiudad(concat(DAYNAME(CURRENT DATE()), NEW.id dep));
    if DIA = true
        then
             signal sqlstate '45000' set message_text = 'NO SE ADMITE INSERCICIONES
LOS MIERCOLES PARA LA PAZ';
    ELSE
             INSERT INTO auditoria persona
                (operation, stamp, userid, hostname, fullname, edad, sexo) SELECT
                  'I', now(), user(),@@hostname,concat(new.nombre,'
',new.apellidos),NEW.edad,NEW.sexo;
    end if ;
END;
insert into PERSONA(id per, nombre, apellidos, fecha nac, edad, email, id dep,
id prov, sexo) values
(123, 'nombre6', 'apellidos6', '1999-10-30', 19, 'nombre6@gmail.com', 2, 5, 'f');
        insert into PERSONA(id_per, nombre, apellidos, fecha_nac, edad, email, id_dep, id_prov, sexo) values
        (123,'nombre6', 'apellidos6', '1999-10-30', 19, 'nombre6@gmail.com', 2, 5,'f');
 1091
  [45000][1644] NO SE ADMITE INSERCICIONES FINES DE SEMANA
```

• Opcional, resolver el ejercicio anterior haciendo que la función solo reciba un parámetro.

```
create function verificaDiaCiudad(diaciudad text)returns boolean
    begin
    declare dia integer default 0;
    declare ciuda integer default 0;
    declare res boolean default false;
    set dia = locate('Wednesday',diaciudad);
    SET ciuda = locate('2',diaciudad);#laPaz es numero 2 en id
    if dia > 0 and ciuda >0
        then
        set res = true;
    end if;
    return res;
    #return if(diaciudad = 'Wednesday2',true,false);
    end;
drop function verificaDiaCiudad;
```