

# Cursos Medicos

Se creara una base de datos para administrar y controlar los diversos cursos que serán implementados por diferentes doctores, abarcando diferentes áreas o especialidades. Estos cursos permitiran a practicantes de medicina, doctores o interesados en general a adquirir conocomientos relacionados al area que le permitan desarrollar habilidades de prevencion y tener nociones sobre dierentes enfermedades, padecimientos y su tratamiento

Esta base de datos contemplara los cursos, doctores y alumnos, asi mismo se llevara control de evaluaciones que permitiran la obtencion de reconocimientos a los participantes.

**Tabla Doctores**

Contiene la informacion relacionada a los doctores o especialistas que impartiran el curso

TYPE KEY	COLUMN	TYPE	NOT NULL	UNIQUE	LENGTH	DESCRIPTION
PK	ID_DOCTOR	INT	TRUE	TRUE	4	ID DEL DOCTOR O ESPECIALISTA
	NOMBRE	VARCHAR	TRUE		70	CONTIENE EL NOMBRE DEL DOCTOR
	APELLIDOS	VARCHAR	TRUE		70	INFORMACION REFERIDA A LOS APELLIDOS DE LOS MEDICOS O ESPECIALISTAS
	TELEFONO	VARCHAR	TRUE		12	TELEFONO DEL MEDICO O ESPECIALISTA
	DIRECCION	VARCHAR			150	DIRECCION DE DOCTOR O ESPECIALISTA
	CORREO	VARCHAR	TRUE		80	CORREO DEL MEDICO O ESPECIALISTA
FK	CEDULA PROFESIONAL	VARCHAR	TRUE		40	CORRESPONDE A LA CEDULA PROFECIONAL DEL ESPECIALISTA
	ID_ESPECIALIDAD	INT	TRUE		4	Clave foranea de tabla especialidad

### Tabla Especialidades

Contiene la informacion relacionada a los doctores o especialistas que impartiran el curso

TYPE KEY	COLUMN	TYPE	NOT NULL	UNIQUE	LENGTH	DESCRIPTION
PK	ID_ESPECIALIDAD	INT	TRUE	TRUE	4	CORRESPONDE A LA CLAVE PRIMARIA DE LA ESPECIALIDAD
	NOMBRE	VARCHAR	TRUE		70	CONTIENE EL NOMBRE DE LA ESPECIALIDAD
	DESCRIPCION	VARCHAR			200	CONTIENE DESCRIPCION DE LA ESPECIALIDAD

### Tabla Alumnos

Contiene la informacion relacionada a los alumnos que tomaran el curso

TYPE KEY	COLUMN	TYPE	NOT NULL	UNIQUE	LENGTH	DESCRIPTION
PK	ID_ALUMNO	INT	TRUE	TRUE	4	ID DEL ALUMNO
	NOMBRE	VARCHAR	TRUE		50	CONTIENE EL NOMBRE DEL ALUMNO
	APELLIDOS	VARCHAR	TRUE		80	INFORMACION REFERIDA A LOS APELLIDOS DE LOS ALUMNOS
	TELEFONO	VARCHAR	TRUE		12	TELEFONO DEL ALUMNO
	DIRECCION	VARCHAR			200	DIRECCION DE ALUMNO
FK	CORREO	VARCHAR	TRUE		80	CORREO DE ALUMNO
	ID_ALUMNO_CURSO	INT	TRUE		4	CORRESPONDE AL ID DEL CURSO AL QUE ESTA INSCRITO EL ALUMNO

### Tabla Cursos

Contiene la informacion de los cursos a implementare

TYPE KEY	COLUMN	TYPE	NOT NULL	UNIQUE	LENGTH	DESCRIPTION
PK	ID_CURSO	INT	TRUE	TRUE	4	ID DE CURSO
	NOMBRE CURSO	VARCHAR	TRUE		70	CONTIENE EL NOMBRE DEL CURSO
	COSTO	INT	TRUE		10	INFORMACION SOBRE EL COSTO DE CURSO
	FEHCA INICIO	DATE	TRUE			INDICA LA FEHCA EN LA QUE INICIARA EL CURSO
	FECHA FIN	DATE	TRUE			INDICA LA FECHA EN LA QUE TERMINA EL CURSO
FK	ID_TEMARIO	INT	TRUE			INDICA EL ID DEL TEMARIO
FK	ID_TURNO	INT	TRUE			INDICA EL ID DEL TURNO

### Tabla Temario

Contiene la informacion relacionada a los temas y modulos de los cursos

TYPE KEY	COLUMN	TYPE	NOT NULL	UNIQUE	LENGTH	DESCRIPTION
PK	ID_TEMARIO	INT	TRUE	TRUE	4	CORRESPONDE A LA CLAVE PRIMARIA DE LA TABLA
	NOMBRE	VARCHAR	TRUE		120	CONTIENE LA CANTIDAD DE MODULOS DEL CURSO
	DESCRIPCION	VARCHAR	TRUE		400	CONTIENE DESCRIPCION DE CADA MODULO

### Tabla Asistencia

Contiene la relacion de asistencia de los alumnos al curso correspondiente

TYPE KEY	COLUMN	TYPE	NOT NULL	UNIQUE	LENGTH	DESCRIPTION
PK	ID_ASISTENCIA	INT	TRUE	TRUE		CORRESPONDE A LA CLAVE PRIMARIA DE LA TABLA
FK	ID_ALUMNO_CURSO	INT	TRUE			CONTIENE AL ID DEL ALUMNO_CURSO
FK	ID_CURSO	INT	TRUE			CONTIENE EL ID DEL CURSO
	FECHA_ASISTENCIA	DATE	TRUE		60	REGISTRA LA FECHA EN QUE SE TOMA LA ASISTENCIA

### Tabla Modalidad de Curso

Contiene modalidad del curso y hace referencia a el tipo de curso(presencial o remoto)

TYPE KEY	COLUMN	TYPE	NOT NULL	UNIQUE	LENGTH	DESCRIPTION
PK	ID_MODALIDAD_CURSO	INT	TRUE	TRUE	4	HACE REFERENCIA AL ID DE LA TABLA
	TIPO_MODALIDAD	VARCHAR	TRUE		50	CONTIENE EL TIPO DE MODALIDAD
FK	ID_CURSO	INT	TRUE			CONTIENE EL ID DE TABLA CURSO

**Tabla Comentarios**

Contiene los comentarios implementados en los diferentes cursos

TYPE KEY	COLUMN	TYPE	NOT NULL	UNIQUE	LENGTH	DESCRIPTION
PK	ID_COMENTARIOS	INT	TRUE	TRUE	4	HACE REFERENCIA AL ID DE LA TABLA
	COMENTARIO	VARCHAR	TRUE		200	ALMACENA LOS COMENTARIOS REALIZADOS POR LOS INVOLUCRADOS EN LOS DIFERENTES CURSOS
FK	ID_CURSO	INT	TRUE		4	CONTIENE EL ID DEL CURSO
	FECHA_COMMENT	TIMESTAMP				CORRESPONDE A LA FECHA EN QUE SE GENERA EL REGISTRO
FK	ID_DOCTOR	INT	TRUE		4	CONTIENE EL ID DEL DOCTOR
FK	ID_ALUMNO	INT	TRUE		4	CONTIENE EL ID DEL ALUMNO

**Tabla Turno**

Contiene los turnos en los que se aplicaran los cursos

TYPE KEY	COLUMN	TYPE	NOT NULL	UNIQUE	LENGTH	DESCRIPTION
PK	ID_TURNO	INT	TRUE	TRUE	4	HACE REFERENCIA AL ID DE LA TABLA
	TURNOS	VARCHAR	TRUE		50	SE REFIERE A LOS TURNOS QUE SE MANEJARAN EN LOS CURSOS

**Tabla Alumno\_Curso**

Contiene los turnos en los que se aplicaran los cursos

TYPE KEY	COLUMN	TYPE	NOT NULL	UNIQUE	LENGTH	DESCRIPTION
PK	ID_ALUMNO_CURSO	INT	TRUE	TRUE	4	HACE REFERENCIA AL ID DE LA TABLA
FK	ID_ALUMNO	VARCHAR	TRUE		4	SE REFIERE A LOS TURNOS QUE SE MANEJARAN EN LOS CURSOS
FK	ID_CURSO	INT	TRUE		4	HACE REFERENCIA AL ID DE LA TABLA CURSO
	CALIFICACION	FLOAT	TRUE		3	HACE REFERENCIA A LA CALIFICACION OBTENIDA EN EL CURSO

Tablas de Logs(ejecucion en base a otra tabla)

Tabla log\_comment

Contiene los registros generados al registrarse un nuevo comentario, esto de realizado por medio de un trigger

TYPE KEY	COLUMN	TYPE	NOT NULL	UNIQUE	LENGTH	DESCRIPTION
PK	LOG_ID	INT	TRUE	TRUE	4	HACE REFERENCIA AL ID DE LA TABLA
FK	ID_COMENTARIO	VARCHAR	TRUE		4	HACE REFERENCIA A LOS COMENTARIO GENERADOS
	USUARIO	VARCHAR	TRUE		4	GUARDA EL DATO DEL USUARIO QUE REALIZA LOS CAMBIOS
	FECHA	DATE			3	FECHA EN QUE SE REALIZA EL REGISTRO
	HORA	TIME				HORA EN QUE SE REALIZA EL REGISTRO

Tabla log\_Calif

Contiene los registros generados al registrarse un nuevo comentario, esto de realizado por medio de un trigger

TYPE KEY	COLUMN	TYPE	NOT NULL	UNIQUE	LENGTH	DESCRIPTION
PK	LOG_ID	INT	TRUE	TRUE	4	HACE REFERENCIA AL ID DE LA TABLA
FK	CALIFICACION	VARCHAR	TRUE		4	A LA CALIFICACION MODIFICADA
	USUARIO	VARCHAR	TRUE		4	GUARDA EL DATO DEL USUARIO QUE REALIZA LOS CAMBIOS
	FECHA	DATE			3	FECHA EN QUE SE REALIZA EL REGISTRO
	HORA	TIME				HORA EN QUE SE REALIZA EL REGISTRO

```
CREATE DATABASE Cursos_Medicos;
USE Cursos_Medicos;
```

```
CREATE TABLE Doctores(
  Id_Doctor int primary key not null auto_increment,
  Nombre varchar(70),
  Apellidos varchar(70),
  Telefono varchar(12),
  Direccion varchar(150),
  Correo varchar(80),
  Cedula_Profesional varchar(40),
  Id_Especialidad int not null
);
```

```
CREATE TABLE Especialidades(
  Id_Especialidad int primary key not null auto_increment,
  Nombre_esp varchar(70),
  Descripcion varchar(150));
```

```
CREATE TABLE Alumnos(
  Id_Alumno int primary key not null auto_increment,
  Nombre varchar(50),
  Apellidos varchar(80),
  Telefono varchar(12),
  Direccion varchar(200),
  Correo varchar(80),
  Id_Alumno_Curso int not null
);
```

```
CREATE TABLE Cursos(
  Id_Curso int primary key not null auto_increment,
  Nombre_Curso varchar(70),
  Costo int,
  Fecha_Inicio date,
  Fecha_Fin date,
  Id_Temario int not null,
  Id_Turno int not null
);
```

```
CREATE TABLE Temario(
  Id_Temario int primary key not null auto_increment,
  Nombre varchar(120),
  Descripcion varchar(150)
);
```

```
CREATE TABLE Asistencia (
  Id_Asistencia int primary key not null auto_increment,
  Fecha_Asistencia datetime,
  Id_Alumno_Curso int not null,
  Id_Curso int not null
);
```

```
CREATE TABLE Modalidad_Curso(
  Id_Modalidad_Curso int primary key not null auto_increment,
```

```
Tipo_Modalidad varchar(50),
Id_Curso int not null
);
```

```
CREATE TABLE Comentarios(
Id_Comentario int primary key not null auto_increment,
Comentario varchar(200),
Id_Curso int not null,
Id_Alumno int,
Id_Doctor int,
fecha_comment timestamp default current_timestamp
);
```

```
CREATE TABLE Turno(
Id_Turno int primary key not null auto_increment,
Turno varchar(50)
);
```

```
CREATE TABLE Alumno_Curso(
Id_Alumno_Curso int primary key not null auto_increment,
calificacion float(10),
Id_curso int not null,
Id_Alumno int not null
);
```

```
CREATE table log_comment(
    log_id int auto_increment primary key not null,
    id_comentario int,
    usuario varchar(100),
    fecha date,
    Hora time
);
```

```
CREATE table logs_Calif(
    log_id int auto_increment primary key not null,
    calificacion int,
    usuario varchar(100),
    fecha date,
    Hora time
);
```

```
insert into especialidades(nombre_esp,descripcion) values (
'Pediatria','Rama de la medicina que se especializa en la salud y las
enfermedades de los niños'),
('Cardiología','Rama de la medicina que se especiliza en el diagnóstico y
tratamiento de enfermedades del corazón, los vasos sanguíneos y el
sistema circulatorio'),
('Enfermeria','abarca el cuidado autónomo y colaborativo de personas de
todas las edades, familias, grupos y comunidades, enfermos o sanos y en
todos los entornos'),
('Medicina Familiar','es la especialidad que brinda atención médica
continua e integral para el individuo y la familia.'),
('Endocrinología','Es una subespecialidad de medicina interna.'),
```

('Neumología','Tratan el sistema cardíaco-pulmonar, consiste en los órganos que trabajan juntos para ayudar a que la respire, como el corazón y los pulmones'),  
 ('Neurología','Los neurólogos son los que tratan a los pacientes con condiciones que afectan al cerebro, espina vertebral o nervios'),  
 ('Patología','La patología estudia las causas y la naturaleza de una enfermedad en particular'),  
 ('Medicina física y rehabilitación','Ayudan a los pacientes con deshabilitación del cerebro, columna vertebral, nervios, huesos, articulaciones, ligamentos, músculos y tendones'),  
 ('Otorrinolaringólogo',' son médicos que se especializan en el diagnóstico y el tratamiento de enfermedades o afecciones del oído, la nariz y la garganta');

insert into

doctores(nombre,apellidos,telefono,direccion,correo,cedula\_profecional,id\_especialidad) values (

'Arturo','Hernandez Reyes','8128564447','santiago 4516, Apodaca, nuevo leon', 'art@gmail.com','568475',1),  
 ('Carlos','Garcia Morales','8125849638','esperanza 25, Monterrey, nuevo leon', 'gmorales@gmail.com','5874210',3),  
 ('Alejandro','Perez Olmos','7823587410','av 20 de noviembre, poza rica veracruz', 'a1985@gmail.com','96854122',5),  
 ('Esteban','Vazquez Santes','7824896523','ciruelos 45 guadalajara jalisco', '@gmail.com','669997712',1),  
 ('Luisa','Escobar Bautista','8128564447','santiago 4516 apodaca nuevo leon', 'luisa@gmail.com','0025484',1),  
 ('ana','Perez Olmos','6925823444','carlos sanz 345, leon guanajuato', 'anapolmos@gmail.com','9852525',4),  
 ('Griselda','Gomez Cruz','2584697485','santiago 4516 queretato queretaro', 'grisgomcruz@gmail.com','44336515',9),  
 ('Fernanado','Martinez Ruiz','6359821479','santiago 4516 san nicolas nuevo leon', 'ferMer@gmail.com','9852614',8),  
 ('Gonzalo','Rocha valdez','7481385410','santiago 4516 apodaca nuevo leon', 'g1979@gmail.com','6659812',10),  
 ('cristobal','Guevara mendez','7325816889','santiago 4516 apodaca nuevo leon', 'cristobal2112@gmail.com','445220',1),  
 ('Carlos','Hernandez Salcedo tovar','5525669102','santiago 4516 apodaca nuevo leon', 'carlos4502@gmail.com','368741',2),  
 ('Maria','Rubalcaba Solis','8124965722','santiago 4516 apodaca nuevo leon', 'mariaRs80@gmail.com','588103',1);

insert into

alumnos(nombre,apellidos,telefono,direccion,correo,id\_alumno\_curso) values (

'javier','balcazar ruelas', 815596004,'mongragon 1256 santiago, nuevo leon', 'monda2213@gmail.com',1),  
 ('ernesto','sanchez martinez', 4489253611,'lopez mateos 09 cd de mexico', 'neto25@gmail.com',2),  
 ('karla','perez vazquez', 112258943,'barnanrdo reyes 2344, anahuac veracruz', 'kpeva@gmail.com',1),  
 ('Martin','perez garcia', 4448009222,'huayaquil 200, nuevo leon', 'martin@gmail.com',1),



```
( 'juan', 'gomez martinez', 5540220397, 'santiago 212
veracruz', 'juan2731@gmail.com', 2),
( 'kevin', 'mar escobar', 5582170002, 'galapagos 404, medrano
sonora', 'k_evin_68@gmail.com', 1),
( 'luis', 'treviño prieto', 9663322441, 'mezquital 23 garcia, nuevo
leon', 'treviño1199@gmail.com', 2),
( 'alan', 'vazquez garza', 8877665544, 'santa rosa 28 4D marquezta, cd
mexico', 'alan4511@gmail.com', 2),
( 'laura', 'solis bueno', 1122334455, 'navidad 201, santa catarina nuevo
leon', 'laurasb@gmail.com', 2);
```

```
insert into cursos(nombre_curso, costo, Fecha_inicio, Fecha_Fin,
id_temario, id_turno) values (
'curso de enfermedades metabolicas', '2500', '2022-05-22', '2022-06-
22', 1, 1),
('Importancia de pruebas cardiacas', '3200', '2022-05-20', '2022-06-
22', 2, 1),
('importancia del tamizaje en los bebes', '2900', '2022-06-04', '2022-07-
02', 3, 2),
('analisis de pruebas y padecimientos en los bebes', '2000', '2022-06-
09', '2022-06-30', 1, 1);
```

```
insert into temario(nombre, descripcion) values (
'definicion y importancia de las pruebas metabolicas', 'se vera las
definiciones y la importancia que tiene este servicio en beneficio de la
salud de los bebes'),
('definicion y consecuencias de pruebas cardiacas', 'se pretende orientar
a los especialistas y a los padres a evaluar las posibilidades de prevenir
la muerte de cuna'),
('importancia del tamizaje en los bebes', 'ver la importancia de un
tamizaje y las consecuencias de las enfermedades a las que se arriesga un
recien nacido'),
('Prueba de eliminacion 1', 'validacion de eliminacion'),
('Prueba de eliminacion 2', 'validacion de eliminacion 2');
```

```
insert into Alumno_Curso(calificacion, id_curso, id_alumno) values(
10, 1, 1),
(9, 1, 2),
(7, 1, 5),
(8, 1, 1),
(10, 1, 7),
(10, 1, 3),
(10, 1, 6);
```

```
insert into comentarios(Comentario, id_curso, Id_alumno, id_doctor)
values(
'me parece que al curso le falta contenido', 1, 4, null),
('es un curso muy completo que cumple con mis requerimientos', 1, 1,
null),
('estoy interesado en cambiar el turno de mi curso', 1, 2, null),
('no me parece que el curso sea apropiado ya que los alumnos no cumplen
con los requerimientos', 1, null, 1),
('soy medico', 1, null, 3),
('soy doctor', 1, null, 5);
```

```
insert into turno(turno)values('Matutino'),('Vespertino'),('Nocturno');
```

```
ALTER TABLE Doctores ADD FOREIGN KEY(Id_Especialidad) REFERENCES  
Especialidades(Id_Especialidad);
```

```
ALTER TABLE Alumnos ADD FOREIGN KEY(Id_Alumno_Curso) REFERENCES  
Alumno_Curso(Id_Alumno_Curso);
```

```
ALTER TABLE Asistencia ADD FOREIGN KEY(Id_Alumno_Curso) REFERENCES  
Alumno_Curso(Id_Alumno_Curso),  
ADD FOREIGN KEY(Id_Curso) REFERENCES Cursos(Id_Curso);
```

```
ALTER TABLE Modalidad_Curso ADD FOREIGN KEY(Id_Curso) REFERENCES  
Cursos(Id_Curso);
```

```
ALTER TABLE Comentarios ADD FOREIGN KEY(Id_Curso) REFERENCES  
Cursos(Id_Curso),  
ADD FOREIGN KEY(Id_Alumno) REFERENCES Alumnos(Id_Alumno),  
ADD FOREIGN KEY(Id_Doctor) REFERENCES Doctores(Id_Doctor) ;
```

```
ALTER TABLE Alumno_Curso ADD FOREIGN KEY(Id_Curso) REFERENCES  
Cursos(Id_Curso),  
ADD FOREIGN KEY(Id_Alumno) REFERENCES Alumnos(Id_Alumno);
```

```
ALTER TABLE Cursos ADD FOREIGN KEY(Id_Temario) REFERENCES  
Temario(Id_Temario),  
ADD FOREIGN KEY(Id_Turno) REFERENCES Turno(Id_Turno);
```

```

use cursos_medicos;

-- SECCION PARA VISTAS
    ## vista que muestra el nombre completo de los
medicos/doctores y la especialidad que tienen cada uno
    CREATE VIEW Medicos_Esp as select d.Nombre, d.Apellidos,
esp.Nombre_esp from doctores d
    join especialidades as esp on esp.Id_Especialidad =
d.Id_Especialidad ;

    select * from medicos_esp; -- query para mostrar vista

    ## vista para mostrar los alumnos inscritos al curso de
enfermedades metabolicas, donde se pueda ver el contacto y correo del
alumno

    create view alumnos_cursos
as select nombre_curso, concat(nombre, ' ', apellidos)
as Nombre_Completo, telefono, correo from alumno_curso ac
    inner join cursos as c on ac.id_curso = c.id_curso
    inner join alumnos as a on a.id_alumno = ac.id_alumno ;

    select * from alumnos_cursos; -- query para mostrar
vista

    ## en esta vista se obtiene la informacion de cada
curso(temario y descripcion)

    create view Datos_Curso as select nombre_curso, nombre
as tema, descripcion from cursos c
    inner join temario as t on t.id_temario = c.id_temario ;

    select * from Datos_Curso; -- query para mostrar vista

    ## vista que muestra la lista de cursos y el costo de cada 1
from cursos;
    create view list_cursos as select nombre_curso, costo

    select * from list_cursos;

    ## vista para mostrar los alumnos que han realizado
comentarios a algun curso y la fecha en que realizaron su comentario
    create view comentarios_cursos as select concat(
nombre, ' ', apellidos) as Nombre_Completo, fecha_comment from alumnos a
    inner join comentarios as c on c.id_alumno = a.id_alumno
;

    select * from comentarios_cursos; -- query para mostrar vista

-- SECCION PARA FUNCIONES
-- FUNCION PARA REALIZAR LA SUMA DEL COSTO DE 2 CURSOS
DELIMITER ##

```

```

CREATE function suma_precio_cursos(precio1 int, precio2 int)
returns int
deterministic
BEGIN
    declare suma_precios int;
    set suma_precios=precio1+precio2;
    return suma_precios;
END ##
select suma_precio_cursos(2500,3200);
drop function name_doctor;

```

```

-- FUNCION PARA MOSTRAR EL NOMBRE DEL MEDICO
DELIMITER ##
CREATE function name_doctor(doctor_user int)
returns varchar (250)
reads sql data
deterministic
BEGIN
    declare datos_name varchar(80);
    select concat(Nombre,',',Apellidos) into datos_name from
cursos_medicos.doctores where Id_Doctor=doctor_user;
    return datos_name;
END ##
select name_doctor(1);

```

```

-- SECCION PARA STORE PROCEDURE
-- STORE PROCEDURE PARA ORDENAMIENTO DE TABLA.- el ordenamiento del store
y se habilita opcion para que el orden pueda ser de manera descendente o
ascendente.

```

```

DELIMITER $$
CREATE PROCEDURE Q_ORDENAMIENTO(column1 varchar(200),orden
INT)
BEGIN
    DECLARE Q_BASE VARCHAR(200);
    DECLARE TIPO_ORDEN VARCHAR(10);
    DECLARE Q_FINAL VARCHAR(200);
    SET Q_BASE = 'SELECT NOMBRE, APELLIDOS, ID_ESPECIALIDAD FROM
cursos_medicos.doctores';

    if orden = 1 THEN
        SET TIPO_ORDEN = 'asc;';
    else
        SET TIPO_ORDEN = 'desc;';
    end if;

    if column1 = "" then
        select 'La columna no puede ser vacia';
    else
        SELECT concat(Q_BASE,' ORDER BY ',column1,' ',
TIPO_ORDEN) into Q_FINAL;
        SET @m_orden = Q_FINAL;

```

```

        PREPARE EJECUTAR FROM @m_orden;
        EXECUTE EJECUTAR;
        deallocate prepare EJECUTAR;
    end if;

    END $$
    DELIMITER ;

    CALL Q_ORDENAMIENTO('Apellidos',1);

    -- STORE PROCEDURE PARA INSERTAR Y ELIMINAR ELEMENTO.- el
    store cuenta con 2 opciones, las cuales dependiendo de la instruccion
    indicada inserta o elimina un registro dentro de la tabla temario

    DELIMITER $$
    CREATE PROCEDURE ACCION_TABLE(
        Accion VARCHAR(200),
        TEM_ID INT,
        NameT varchar(100),
        Desct varchar(100)
    )
    BEGIN
        -- INSERT
        if Accion = "INSERT" THEN
            INSERT INTO Temario(id_temario,Nombre,
Descripcion)
                VALUES (TEM_ID,NameT, Desct);
        end if;
        -- DELETE
        if Accion = "DELETE" THEN
            DELETE FROM Temario
            WHERE Id_temario = TEM_ID;
        end if;
    END$$

    CALL ACCION_TABLE("INSERT",0,'PROBAR INSERT','PROBAR INSERT');
    -- TABLA PARA VALIDAR MODIFICACIONES
    select *from temario;

    -- SECCION PARA TRIGGERS
    -- TRIGGERS PARA VALIDAR E INSERTAR REGISTRO DE COMENTARIOS
    -- tabla para log de comentarios

    -- se genera trigger para validar que campo de comentario no
    sea vacio en caso de que el campo tenga contenido se genera la insercion
    DELIMITER $$
    CREATE TRIGGER valida_comment
    BEFORE INSERT
    ON comentarios FOR EACH ROW
    BEGIN
        DECLARE MSG_ERROR VARCHAR(70);
        if (NEW.comentario = '') then

```

```

        SET MSG_ERROR = 'EL COMENTARIO NO PUEDE SER
VACIO';

        SIGNAL SQLSTATE'45000' SET message_text=
MSG_ERROR;

        end if;
    END$$
DELIMITER ;

-- el trigger insert permite agregar un registro a la tabla
log_comment cada que se ingresa un registro para la tabla comentarios. En
esta tabla se agrega usuario que realiza insercion, fecha y hora
DELIMITER $$
CREATE TRIGGER insert_commetarios
AFTER insert
ON comentarios FOR EACH ROW
BEGIN
    insert into log_comment(id_comentario, usuario, fecha, hora)
values(1, session_user(),current_date(), current_time());
END$$
DELIMITER ;

-- query para insertar registro en tabla comnetarios
insert into comentarios(Comentario, id_curso, Id_alumno,
id_doctor) values('Prueba de comentario', 1, 4, 0);

select * from comentarios; -- query para ver comentarios
ingresados
select * from log_comment; -- query para ver los registros en
log_comment realidos en la tabla comentarios

-- TRIGGERS PARA VALIDAR Y ACTUALIZAR CALIFICACION DE ALUMNOS
-- tabla para log donde se guardaran los datos de quien
modifica la fecha y la hora en que se realiza un cambio

-- de declara triger que valida que se ingrese una calificacion
permitida para hacer la actualizacion
DELIMITER $$
CREATE TRIGGER valida_calificacion
BEFORE update
ON Alumno_Curso FOR EACH ROW
BEGIN
    DECLARE MSG_ERROR VARCHAR(70);
    if (NEW.calificacion = 0) then
        SET MSG_ERROR = 'Ingrese una calificacion valida';
        SIGNAL SQLSTATE'45000' SET message_text=
MSG_ERROR;
    end if;
END$$
DELIMITER ;

-- de declara triger para que al actualizar un registro de
calificacion, se genere un log del cambio realizado con los datos
correspondientes a quien realiza el cambio o actualizacion
DELIMITER $$
CREATE TRIGGER update_calificacion

```

```

        AFTER update
        ON Alumno_Curso FOR EACH ROW
        BEGIN
            insert into logs_Calif(calificacion, usuario, fecha,
hora) values(1, session_user(),current_date(), current_time());
        END$$
        DELIMITER ;

-- query para actualizar calificacion de tabla
update Alumno_Curso
set calificacion = 7,
    id_curso = 1,
    id_alumno = 1
where id_alumno_curso = 1;

select * from Alumno_curso; -- para mostrar tabla
select * from logs_calif;    -- para mostrar tabla donde se
guardan los logs del cambio

```