



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
SISTEMAS DE BASES DE DATOS

Integrantes:

Jorge Gabriel Caicedo Viteri
Anthony Esteven Quishpe Guaytarilla

Procedimientos

tb_persona	
pk	int_index_persona str_id_persona str_nombre_persona str_apellido_persona str_fecha_nacimiento_persona int_edad_persona

tb_persona
int_index_persona
str_id_persona
str_nombre_persona
str_apellido_persona
str_fecha_nacimiento_persona
int_edad_persona

¿Que almacena la tabla Persona?

Almacena lo que es el Id, Nombre, Apellido, Fecha de nacimiento y edad de la persona ya sea un estudiante, docente o representante del estudiante.

Insert:

No podemos añadir una nueva persona mediante la interfaz de usuario porque no se poseen los permisos para dicha acción. Si se buscara añadir una nueva persona debería ser mediante un administrador o algún usuario con permisos para dicha acción.

```
SQL> create or replace procedure "sp_CrearPersona" (int_index_persona in number, str_id_persona in varchar2,  
2 str_nombre_persona in varchar2, str_apellido_persona in varchar2, str_fecha_nacimiento_persona in varchar2)  
3 AS  
4 Begin  
5     INSERT INTO TB_PERSONA(INT_INDEX_PERSONA, STR_ID_PERSONA, STR_NOMBRE_PERSONA, STR_APELLIDO_PERSONA, STR_FECHA_NACIM  
INTO_PERSONA)  
6     VALUES(int_index_persona, str_id_persona,  
7     str_nombre_persona, str_apellido_persona, str_fecha_nacimiento_persona);  
8 End;  
9 /  
Procedimiento creado.
```

```
SQL> exec "sp_CrearPersona" (1, '1725497190', 'Gabriel', 'Caicedo', '26-09-2000');  
Procedimiento PL/SQL terminado correctamente.
```

```
SQL> select INT_INDEX_PERSONA, STR_ID_PERSONA, STR_NOMBRE_PERSONA, STR_APELLIDO_PERSONA, STR_FECHA_NACIMIENTO_PERSONA from tb_persona;  
INT_INDEX_PERSONA STR_ID_PER STR_NOMBRE STR_APELLI STR_FECHA_ A  
-----  
1 1725497190 Gabriel Caicedo 26-09-2000 Ve
```

Update:

No podemos actualizar la información personal de cada persona, porque ya están definidos en el registro civil y no nos compete actualizar algún dato de cada persona.

Delete:

No podemos borrar el registro de una persona mediante una interfaz de usuario porque no se poseen los permisos para dicha acción. Si se buscara eliminar alguna



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

SISTEMAS DE BASES DE DATOS

persona debería ser mediante un administrador o algún usuario con permisos para dicha acción.

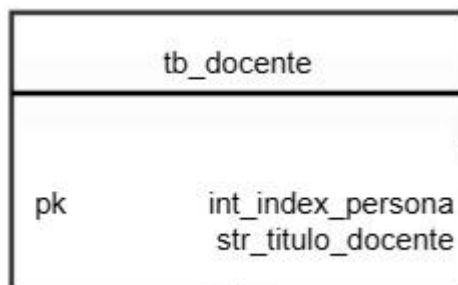
```
SQL> CREATE OR REPLACE PROCEDURE "sp_BorrarPersona" (int_index_persona in number)
  2 AS
  3 BEGIN
  4     DELETE FROM TB_PERSONA WHERE INT_INDEX_PERSONA = int_index_persona;
  5 END;
  6 /
```

Procedimiento creado.

```
SQL> exec "sp_BorrarPersona" (1);
```

Procedimiento PL/SQL terminado correctamente.

```
SQL> select INT_INDEX_PERSONA,STR_ID_PERSONA,STR_NOMBRE_PERSONA,STR_APELLIDO_PERSONA,STR_FECHA_NACIMIENTO_PERSONA from tb_persona;
ninguna fila seleccionada
```



tb_docente

int_index_persona
str_titulo_docente

¿Que almacena la tabla Docente?

Almacena el id y el título que posee el docente, mencionando que la id también se debe encontrar en la tabla Persona.

Insert:

Si podemos añadir un nuevo docente mediante la interfaz de usuario porque si la id del nuevo docente se encuentra en la tabla Persona se lo podría añadir con su respectivo título.

```
SQL> create or replace procedure "sp_CrearDocente" (int_index_persona in number,str_titulo_docente in varchar2)
  2 AS
  3 Begin
  4     INSERT INTO TB_DOCENTE(INT_INDEX_PERSONA,STR_TITULO_DOCENTE)
  5     VALUES(int_index_persona,str_titulo_docente);
  6 End;
  7 /
```

Procedimiento creado.

```
SQL> exec "sp_CrearDocente" (1,'Ingeniero');
```

Procedimiento PL/SQL terminado correctamente.

```
SQL> select INT_INDEX_PERSONA,STR_TITULO_DOCENTE from tb_docente;
```

```
INT_INDEX_PERSONA STR_TITULO
-----
1 Ingeniero
```



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
SISTEMAS DE BASES DE DATOS

Update:

Solo se puede actualizar el título que posee el Docente porque puede darse el caso de que el Docente cambie su título académico, el identificador no se podría porque es un atributo heredado.

```
SQL> CREATE OR REPLACE PROCEDURE "sp_ActualizarDocente" (int_index_persona in number, str_titulo in varchar2)
  2 AS
  3 Begin
  4     UPDATE TB_DOCENTE
  5     SET
  6     STR_TITULO_DOCENTE = str_titulo
  7     WHERE INT_INDEX_PERSONA = int_index_persona;
  8 END;
  9 /
Procedimiento creado.
```

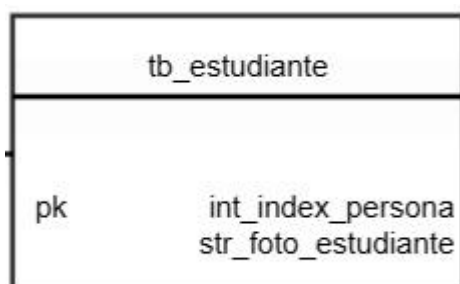
```
SQL> exec "sp_ActualizarDocente" (1,'Doctor');
Procedimiento PL/SQL terminado correctamente.
SQL> select INT_INDEX_PERSONA,STR_TITULO_DOCENTE from tb_docente;
INT_INDEX_PERSONA STR_TITULO
-----
1 Doctor
```

Delete:

Podemos borrar los datos completos del docente, porque puede que el tiempo de contrato ya haya caducado y en ese caso se borrarán los datos completos del docente, pero no podemos borrar solo el título o el identificador.

```
SQL> CREATE OR REPLACE PROCEDURE "sp_BorrarDocente" (int_index_persona in number)
  2 AS
  3 BEGIN
  4     DELETE FROM TB_DOCENTE WHERE INT_INDEX_PERSONA = int_index_persona;
  5 END;
  6 /
Procedimiento creado.
```

```
SQL> exec "sp_BorrarDocente" (1);
Procedimiento PL/SQL terminado correctamente.
SQL> select INT_INDEX_PERSONA,STR_TITULO_DOCENTE from tb_docente;
ninguna fila seleccionada
```



tb_estudiante
int_index_persona
str_foto_estudiante



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
SISTEMAS DE BASES DE DATOS

¿Que almacena la tabla Estudiante?

La tabla estudiante almacena el identificador y la respectiva foto de cada estudiante.

Insert:

Si podemos crear nuevos estudiantes mediante una interfaz porque en cada año escolar siempre existe la posibilidad del ingreso de nuevos estudiantes.

```
SQL> create or replace procedure "sp_CrearEstudiante" (int_index_persona in number,str_foto_estudiante in varchar2)
  2 AS
  3 Begin
  4     INSERT INTO TB_ESTUDIANTE(INT_INDEX_PERSONA,STR_FOTO_ESTUDIANTE)
  5     VALUES(int_index_persona,str_foto_estudiante);
  6 End;
  7 /
Procedimiento creado.
```

```
SQL> exec "sp_CrearEstudiante" (1,'Gabriel1roBasico.png');
Procedimiento PL/SQL terminado correctamente.
```

```
SQL> select INT_INDEX_PERSONA,STR_FOTO_ESTUDIANTE from tb_estudiante;

INT_INDEX_PERSONA STR_FOTO_ESTUDIANTE
-----
1 Gabriel1roBasico.png
```

Update:

Solo se puede actualizar la foto porque puede darse el caso de que el estudiante quiera cambiar la foto que lo identifique, el identificador no se podría porque es un atributo heredado.

```
SQL> CREATE OR REPLACE PROCEDURE "sp_ActualizarEstudiante" (int_index_persona in number,str_foto in varchar2)
  2 AS
  3 Begin
  4     UPDATE TB_ESTUDIANTE
  5     SET
  6     STR_FOTO_ESTUDIANTE = str_foto
  7     WHERE INT_INDEX_PERSONA = int_index_persona;
  8 END;
  9 /
Procedimiento creado.
```

```
SQL> exec "sp_ActualizarEstudiante" (1,'Gabriel8voBasica.png');
Procedimiento PL/SQL terminado correctamente.

SQL> select INT_INDEX_PERSONA,STR_FOTO_ESTUDIANTE from tb_estudiante;

INT_INDEX_PERSONA STR_FOTO_ESTUDIANTE
-----
1 Gabriel8voBasica.png
```



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
SISTEMAS DE BASES DE DATOS

Delete:

Si se puede borrar el registro de un estudiante porque puede darse el caso de que el estudiante ya no se encuentre vinculado con la institución por más de 2 años y su información ya sería irrelevante almacenarla.

```
SQL> CREATE OR REPLACE PROCEDURE "sp_BorrarEstudiante" (int_index_persona in number)
2 AS
3 BEGIN
4     DELETE FROM TB_ESTUDIANTE WHERE INT_INDEX_PERSONA = int_index_persona;
5 END;
6 /
```

Procedimiento creado.

```
SQL> exec "sp_BorrarEstudiante" (1);
```

Procedimiento PL/SQL terminado correctamente.

```
SQL> select INT_INDEX_PERSONA,STR_FOTO_ESTUDIANTE from tb_estudiante;
ninguna fila seleccionada
```

tb_representante_estudiante	
pk,fk	int_index_persona_estudiante
pk,fk	int_index_persona_representante
	str_parentesco

tb_representantes_estudiante

int_index_persona_estudiante

int_index_persona_representante

¿Que almacena la tabla Representante Estudiante?

La tabla de los representantes de los estudiantes almacena los identificadores tanto del estudiante como del representante, almacenando también el parentesco entre ambos.

Insert:

Si podemos vincular a nuevos representantes con el respectivo estudiante siempre y cuando ya no se encuentre un registro similar porque siempre puede darse el caso que una nueva persona pase a ser el representante legal de algún estudiante.

```
SQL> create or replace procedure "sp_CrearRepresentante" (int_index_persona_estudiante in NUMBER,int_index_persona_representante in NUMBER,str_parentesco in VARCHAR2)
2 AS
3 Begin
4     INSERT INTO TB_REPRESENTANTE_ESTUDIANTE(INT_INDEX_PERSONA_ESTUDIANTE,INT_INDEX_PERSONA_REPRESENTANTE,STR_PARENTESCO)
5     VALUES(int_index_persona_estudiante,int_index_persona_representante,str_parentesco);
6 End;
7 /
```

Procedimiento creado.

```
SQL> exec "sp_CrearRepresentante" (2,1,'Padre');
```

Procedimiento PL/SQL terminado correctamente.

```
SQL> select INT_INDEX_PERSONA_ESTUDIANTE,INT_INDEX_PERSONA_REPRESENTANTE,STR_PARENTESCO from tb_representante_estudiante;

INT_INDEX_PERSONA_ESTUDIANTE INT_INDEX_PERSONA_REPRESENTANTE STR_PAR
-----
2                             1 Padre
```

Update:



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

SISTEMAS DE BASES DE DATOS

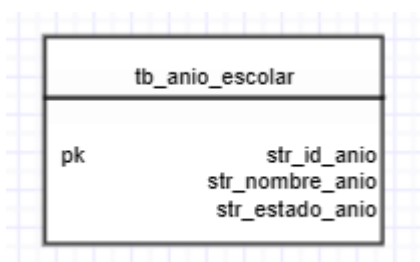
No se podría actualizar algún registro de esta tabla porque tanto las ids relacionadas como el parentesco no van a cambiar en el paso del tiempo, para esto existe como mejor alternativa el Insert.

Delete:

Si se puede borrar un registro de esta tabla porque como se mencionaba puede darse el caso de que una persona deje de representar a algún estudiante.

```
SQL> CREATE OR REPLACE PROCEDURE "sp_BorrarRepresentante" (int_index_persona_estudiante in NUMBER,int_index_persona_representante in NUMBER)
2 AS
3 BEGIN
4 DELETE FROM TB_REPRESENTANTE_ESTUDIANTE
5 WHERE INT_INDEX_PERSONA_ESTUDIANTE = int_index_persona_estudiante
6 AND INT_INDEX_PERSONA_REPRESENTANTE = int_index_persona_representante;
7 END;
8 /
Procedimiento creado.
```

```
SQL> exec "sp_BorrarRepresentante" (2,1);
Procedimiento PL/SQL terminado correctamente.
SQL> select INT_INDEX_PERSONA_ESTUDIANTE,INT_INDEX_PERSONA_REPRESENTANTE,STR_PARENTESCO from tb_representante_estudiante;
ninguna fila seleccionada
```



tb_anio_escolar

str_id_anio
str_nombre_anio
str_estado_anio

¿Que almacena la tabla Año Escolar?

La entidad Año Escolar almacena, lo que respecta al año escolar que corresponda y tenemos como campos el identificador, el nombre y estado del año escolar.

Insert:

Si se puede crear, puesto que el año escolar va cambiando y por ende se tiene que crear un nuevo año escolar cada determinado tiempo.

```
SQL> create or replace procedure "sp_CrearAnio" (str_id_anio in varchar2, str_nombre_anio in varchar2, str_estado_anio in varchar2)
2 AS
3 Begin
4 INSERT INTO "TB_ANIO_ESCOLAR" VALUES(str_id_anio, str_nombre_anio, str_estado_anio);
5 End;
6 /
Procedimiento creado.
```

```
SQL> exec "sp_CrearAnio" ('A2022','Ejemplo','Activo');
Procedimiento PL/SQL terminado correctamente.
```



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
SISTEMAS DE BASES DE DATOS

STR_ID_ANIO	STR_NOMBRE_ANIO	STR_ESTADO
A2025	Ejemplo	Activo
A2022	Ejemplo	Activo
A2023	Ejemplo	Activo
A2024	Ejemplo	Activo

SQL>

Update:

Podemos actualizar el campo Estado del año escolar porque, cada que termina el periodo escolar el estado del año, pasa de activo a inactivo.

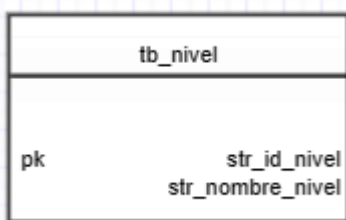
```
SQL> CREATE OR REPLACE PROCEDURE "sp_ActualizarAnio" (VarIdAnio in varchar2, VarNombreAnio in varchar2, VarEstadoAnio in varchar2)
2 AS
3 BEGIN
4     UPDATE TB_ANIO_ESCOLAR
5     SET
6         str_nombre_anio = VarNombreAnio,
7         str_estado_anio = VarEstadoAnio
8     WHERE str_id_anio = VarIdAnio;
9 END;
10 /
Procedimiento creado.
```

```
SQL> exec "sp_ActualizarAnio" ('A2022','Ejemplo','Inactivo');
Procedimiento PL/SQL terminado correctamente.
```

STR_ID_ANIO	STR_NOMBRE_ANIO	STR_ESTADO
A2025	Ejemplo	Activo
A2022	Ejemplo	Inactivo
A2023	Ejemplo	Activo
A2024	Ejemplo	Activo

Delete:

No podemos eliminar el año escolar, porque al querer un estudiante visualizar las materias recibidas al realizar un delete cada que se acabe un periodo no podrá visualizar nada de lo que recibió, para evitar el delete existe el update del estado.



tb_nivel
str_id_nivel
str_nombre_nivel

¿Que almacena la tabla Nivel?



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

SISTEMAS DE BASES DE DATOS

Lo que almacena la entidad Nivel, hace referencia al nivel en el que se encuentra un estudiante al inscribirse en el sistema educativo y como campos a almacenar tenemos el identificador y el nombre del nivel.

Insert:

Se podrían añadir niveles de acuerdo a la base legal ya que por el momento la LOEI para la educación de Infantes y Educación Básica trabaja desde los niveles de Inicial hasta 3ro de Bachillerato, sin embargo, dichos niveles puede darse el caso de añadir algún nivel más en caso de que así lo dicte la LOEI la cual es la base legal.

```
SQL> create or replace procedure "sp_CrearNivel" (str_id_nivel in varchar2, str_nombre_nivel in varchar2)
  2 AS
  3 Begin
  4     INSERT INTO "TB_NIVEL" VALUES(str_id_nivel, str_nombre_nivel);
  5 End;
  6 /

Procedimiento creado.

SQL> exec "sp_CrearNivel" ('Niv004','Nivel 3');

Procedimiento PL/SQL terminado correctamente.
```

STR_ID_NIVEL	STR_NOMBRE_NIVEL
Niv004	Nivel 3
Pru001	Nivel 3
Niv001	Nivel 1
Niv002	Nivel 2
Niv003	Nivel 3

Update:

Si se pueden actualizar los niveles porque deben ser acordes a la base legal LOEI ya que se puede dar el caso de que algún nivel cambie de nombre .

```
SQL> CREATE OR REPLACE PROCEDURE "sp_ActualizarNivel" (VarIdNivel IN VARCHAR2, VarNombreNivel IN VARCHAR2)
  2 AS
  3 BEGIN
  4     UPDATE "TB_NIVEL"
  5     SET
  6         STR_NOMBRE_NIVEL = VarNombreNivel
  7     WHERE STR_ID_NIVEL = VarIdNivel;
  8 END;
  9 /

Procedimiento creado.

SQL> set autoprint on;
SQL> exec "sp_ActualizarNivel" ('Niv001','Nivel 4');

Procedimiento PL/SQL terminado correctamente.
```




UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

SISTEMAS DE BASES DE DATOS

STR_ID_NIVEL	STR_NOMBRE_NIVEL
Niv004	Nivel 3
Pru001	Nivel 3
Niv001	Nivel 4
Niv002	Nivel 2
Niv003	Nivel 3

Delete:

No se pueden borrar los niveles porque ya están definidos por la institución educativa y la LOEI, además los niveles se encuentran en las matrículas en todo el tiempo del funcionamiento de nuestra base de datos. Si se buscase eliminar un nivel este no debería estar vinculado a ningún registro de las tablas hijas de la tabla Nivel.

tb_matricula	
pk,fk	int_index_persona
pk,fk	str_id_anio
fk	str_id_nivel
	str_estado

tb_matricula

int_index_persona
str_id_anio
str_id_nivel
str_estado

¿Que almacena la tabla Matricula?

En esta entidad almacenamos los identificadores del año escolar, la persona es decir el estudiante y el nivel, también posee un campo que pertenece al estado de la matrícula del estudiante.

Insert:

Si podemos crear nuevos registros de matrículas porque en cada año escolar siempre van a aparecer nuevas matrículas donde variarán el año en el que se matricula y el nivel al que va el estudiante.

```
SQL> create or replace procedure "sp_CrearMatricula" (int_index_persona NUMBER, str_id_anio VARCHAR2, str_id_nivel VARCHAR2,
2 str_estado VARCHAR2)
3 AS
4 Begin
5     INSERT INTO TB_MATRICULA VALUES(int_index_persona, str_id_anio, str_id_nivel, str_estado);
6 End;
7 /

Procedimiento creado.

SQL> exec "sp_CrearMatricula" (2, 'A2022', 'Niv004', 'Activo');

Procedimiento PL/SQL terminado correctamente.
```

```
SQL> Select * from tb_matricula;

INT_INDEX_PERSONA STR_ID_ANIO STR_ID_NIVEL STR_ESTA
-----
2 A2022 Niv004 Activo
```

Update:



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

SISTEMAS DE BASES DE DATOS

Lo único que podemos actualizar dentro de esta entidad es el Estado de la matrícula, porque un estudiante puede ser que al principio esté inscrito pero después se dé de baja en ese caso el estado de la matrícula pasa de activo a inactivo.

```
SQL> CREATE OR REPLACE PROCEDURE "sp_ActualizarMatricula" (int_index_persona NUMBER, str_id_anio VARCHAR2, VarEstado VARCHAR2)
  2 AS
  3 BEGIN
  4     UPDATE TB_Matricula
  5     SET
  6     STR_Estado = VarEstado
  7     WHERE INT_INDEX_PERSONA = int_index_persona
  8     AND STR_ID_ANIO = str_id_anio;
  9 END;
 10 /

Procedimiento creado.

SQL> exec "sp_ActualizarMatricula" (2,'A2022','Inactivo');

Procedimiento PL/SQL terminado correctamente.

SQL> Select * from tb_matricula;

INT_INDEX_PERSONA STR_ID_ANIO STR_ID_NIVEL STR_ESTA
-----
                2 A2022      Niv004      Inactivo
```

Delete:

No podemos borrar ningún registro de matrícula porque tanto la LOEI como la AMIE piden un informe respectivo de todos los estudiantes matriculados por año escolar ya sea que el estudiante estuvo activo durante todo el año o no.

tb_materia	
pk	str_id_materia
fk	str_id_nivel
	str_nombre_materia
	str_descripcion_materia

tb_materia
str_id_materia
str_id_nivel
str_nombre_materia
str_descripcion_materia

¿Que almacena la tabla Materia?

Almacena todo lo que tiene que ver con la materia es decir, almacena el identificador, el nombre y una descripción de la materia, además tenemos un campo heredado es el identificador de la entidad Nivel.

Insert:

Si se pueden crear nuevas materias con sus respectivas descripciones e identificadores, mencionando que la id del nivel como es un atributo heredado sólo se podrá realizar el insert siempre y cuando la id del nivel se encuentre en la tabla Nivel.

```
SQL> create or replace procedure "sp_CrearMateria" (str_id_materia in varchar2, str_nombre_materia in varchar2, str_descripcion_materia in varchar2, str_id_nivel in var
char2)
  2 AS
  3 Begin
  4     INSERT INTO "TB_MATERIA" VALUES(str_id_materia, str_nombre_materia, str_descripcion_materia, str_id_nivel);
  5 End;
  6 /

Procedimiento creado.

SQL> exec "sp_CrearMateria" ('Mat001','Algebra','Ciencias Exactas','Pru001');
```



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
SISTEMAS DE BASES DE DATOS

STR_ID_MATERIA	STR_NOMBRE_MATERIA	STR_DESCRI	STR_ID_NIV
Mat001	Algebra	Ciencias E	Pru001 xactas
Mat002	Quimica	Ciencias E	Niv001 xactas
Mat003	Programacion	Computacio	Niv002 n
Mat004	Matematicas	Ciencias E	Niv003 xactas

Update:

Si se puede actualizar las materias porque se puede actualizar el nombre de la materia o la descripción de la misma, además si se da el caso podemos cambiar el nivel al que se encuentra.

```
SQL> CREATE OR REPLACE PROCEDURE "sp_ActualizarMateria" (VarIdMateria in varchar2, VarNombreMateria in varchar2, VarDescripcionMateria in varchar2, VarIdNivel in varchar2)
2 AS
3 BEGIN
4     UPDATE "TB_MATERIA"
5     SET
6         STR_NOMBRE_MATERIA = VarNombreMateria,
7         STR_DESCRIPCION_MATERIA = VarDescripcionMateria,
8         STR_ID_NIVEL = VarIdNivel
9     WHERE STR_ID_MATERIA = VarIdMateria;
10 END;
11 /
Procedimiento creado.
SQL> set autoprint on;
SQL> exec "sp_ActualizarMateria" ('Mat002','Quimica','Ciencias Quimicas','Niv001');
Procedimiento PL/SQL terminado correctamente.
```

Activar Windows
Ve a Configuración para activar Windows.

STR_ID_MATERIA	STR_NOMBRE_MATERIA	STR_DESCRI	STR_ID_NIV
Mat001	Algebra	Ciencias E	Pru001 xactas
Mat002	Quimica	Ciencias Q	Niv001 uimicas
Mat003	Programacion	Computacio	Niv002 n
Mat004	Matematicas	Ciencias E	Niv003 xactas

Delete:

No podemos borrar alguna materia ya que puede darse el caso de que existan calificaciones relacionadas a dicha materia en la tabla calificaciones. Si se buscase eliminar alguna materia esta no debería encontrarse en ningún registro de la tabla de calificaciones ya que la base legal LOEI también menciona que se deben tener los promedios o calificaciones de los estudiantes en las materias impartidas.

tb_actividad	
pk	str_id_actividad
pk, fk	str_id_materia
	str_tema_actividad
	str_estado_actividad

tb_actividad

str_id_actividad
str_id_materia
str_tema_actividad
str_estado_actividad



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
SISTEMAS DE BASES DE DATOS

¿Que almacena la tabla Actividad?

Lo que almacena la entidad actividad son las ids tanto de la materia a la que pertenece como la id de la actividad, los temas de una tarea propuesta y el estado de la misma.

Insert:

Podemos crear nuevas tareas o actividades y se puede porque cada clase el docente enviará una tarea y por ende en el sistema se puede ir creando la nueva actividad, mencionando que solo se permite la creación de la actividad siempre y cuando exista la id de la materia relacionada.

```
SQL> create or replace procedure "sp_CrearActividad" (str_id_actividad in varchar2, var_tema_actividad in varchar2, str_estado_actividad in varchar2, str_id_materia in
varchar2)
2 AS
3 Begin
4     INSERT INTO "TB_ACTIVIDAD" VALUES(str_id_actividad, var_tema_actividad, str_estado_actividad, str_id_materia);
5 End;
6 /
Procedimiento creado.
SQL> exec "sp_CrearActividad" ('Act001','Deber','Activo','Mat002');
```

STR_ID_ACTIVIDAD	VAR_TEMA_ACTIVIDAD	STR_ESTADO	STR_ID_MATERIA
Act001	Deber	Activo	Mat002

Update:

Podemos actualizar los campos de la actividad, como el tema y el estado, en tema puede que se cambie la actividad y el estado puede pasar de activo a inactivo cuando se lo culmine lo que no podemos actualizar son los identificadores de la actividad y de la materia.

```
SQL> CREATE OR REPLACE PROCEDURE "sp_ActualizarActividad" (VarIdActividad in varchar2, VarTemaActividad in varchar2, VarEstadoAc in varchar2, VarIdMateria in varchar2)
2 AS
3 BEGIN
4     UPDATE TB_ACTIVIDAD
5     SET
6         var_tema_actividad = VarTemaActividad,
7         str_estado_actividad = VarEstadoAc,
8         str_id_materia = VarIdMateria
9     WHERE str_id_actividad = VarIdActividad;
10 END;
11 /
Procedimiento creado.
SQL> set autoprnt on;
SQL> exec "sp_ActualizarActividad" ('Act001','Trabajo','Inactivo','Mat002');
```

Activar Windows
Ve a Configuración para activar Windows.

```
SQL> select * from tb_actividad;
```

STR_ID_ACTIVIDAD	VAR_TEMA_ACTIVIDAD	STR_ESTADO	STR_ID_MATERIA
Act001	Trabajo	Inactivo	Mat002

```
SQL> |
```

Delete:



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

SISTEMAS DE BASES DE DATOS

No se puede borrar la actividad en caso de que ya se encuentre registrada alguna calificación a dicha actividad por lo que se mencionaba anteriormente que la LOEI pide las calificaciones de los estudiantes, sin embargo, si se buscase eliminar dicha actividad esta no se debería encontrar en ningún registro de calificación.

tb_calificaciones	
pk,fk	str_id_actividad
pk,fk	str_id_materia
pk,fk	str_id_libreta
	float_nota

tb_calificaciones

str_id_actividades

str_id_materia

str_id_libreta

float_nota

¿Que almacena la tabla Calificaciones?

La tabla de Calificaciones se crea por la relación que existe entre Actividad y Calificación y los campos a almacenar son los identificadores de materia, actividad, calificación y la respectiva nota donde se registraran las calificaciones de los estudiantes por las actividades realizadas en las diferentes materias como lo pide la base legal LOEI.

Insert:

Podemos crear un nuevo registro porque para cada actividad que se realice esta deberá ser calificada a cada estudiante vinculado a la misma.

```
SQL> create or replace procedure "sp_CrearCalificaciones" (str_id_materia in varchar2, str_id_actividad in varchar2, str_id_libreta in varchar2, float_nota in number)
2 AS
3 Begin
4   INSERT INTO "TB_CALIFICACIONES" VALUES(str_id_materia, str_id_actividad, str_id_libreta, float_nota);
5 End;
6 /
Procedimiento creado.
SQL> exec "sp_CrearCalificaciones" ('Mat002','Act001','Lib001', 18);
```

```
SQL> select * from tb_calificaciones;
```

STR_ID_MATERIA	STR_ID_ACTIVIDAD	STR_ID_LIB	FLOAT_NOTA
Mat002	Act001	Lib001	18

```
SQL> |
```

Update:

Lo único que podemos actualizar dentro esta entidad es la nota, porque puede que se haya realizado mal la calificación entonces se puede actualizar.



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
SISTEMAS DE BASES DE DATOS

```
SQL> CREATE OR REPLACE PROCEDURE "sp_ActualizarCalificaciones" (VarIdMateria in varchar2, VarIdActividad in varchar2, VarIdLibreta in varchar2, FloatNota in number)
2 AS
3 BEGIN
4     UPDATE "TB_CALIFICACIONES"
5     SET
6         str_id_actividad = VarIdActividad,
7         str_id_libreta = VarIdLibreta,
8         float_nota = FloatNota
9     WHERE str_id_materia = VarIdMateria;
10 END;
11 /
Procedimiento creado.
SQL> set autprint on;
SQL> exec "sp_ActualizarCalificaciones" ('Mat002','Act001','Lib001', 15);
Procedimiento PL/SQL terminado correctamente.
```

Activar Windows
Ve a Configuración para activar Windows.

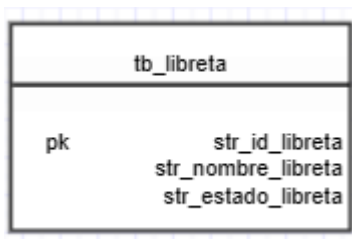
```
SQL> select * from tb_calificaciones;
```

STR_ID_MATERIA	STR_ID_ACTIVIDAD	STR_ID_LIB	FLOAT_NOTA
Mat002	Act001	Lib001	15

```
SQL> |
```

Delete:

No podemos borrar ningún registro de nuestra entidad, porque todos los registros sirven para sacar los promedios por materia como lo dicta la LOEI. En caso de que se busque eliminar alguna calificación solo podrá ser realizada por algún usuario administrador.



tb_libreta

str_id_libreta
str_nombre_libreta
str_estado_libreta

¿Que almacena la tabla Libreta?

La tabla libreta almacena el identificador, el nombre y el estado de las diversas libretas que se encontrarán por estudiante en cada año escolar.

Insert:

Si podemos crear nuevas libretas porque por cada matrícula se debe tener la libreta respectiva al estudiante matriculado referenciando al estudiante como al año escolar que pertenece la libreta.

```
SQL> create or replace procedure "sp_CrearLibreta" (str_id_libreta in varchar2, str_nombre_libreta in varchar2, str_estado_libreta in varchar2)
2 AS
3 Begin
4     INSERT INTO "TB_LIBRETA" VALUES(str_id_libreta, str_nombre_libreta, str_estado_libreta);
5 End;
6 /
Procedimiento creado.
SQL> exec "sp_CrearLibreta" ('Lib001','Calificaciones','Activo');
```




UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
SISTEMAS DE BASES DE DATOS

```
SQL> COLUMN str_id_libreta FORMAT A20
SQL> COLUMN str_nombre_libreta FORMAT A20
SQL> COLUMN str_estado_libreta FORMAT A10
SQL>
SQL> select * from tb_libreta;

STR_ID_LIBRETA          STR_NOMBRE_LIBRETA      STR_ESTADO
-----
Lib001                  Calificaciones          Activo

SQL>
```

Update:

Solo se puede actualizar el estado de la libreta ya que una vez se acabe el año escolar esta pasará de en proceso a finalizada ya que se culminó el año escolar.

```
SQL> CREATE OR REPLACE PROCEDURE "sp_ActualizarLibreta" (VarIdLibreta in varchar2, VarNombreLibreta in varchar2, VarEstadoLibreta in varchar2)
2 AS
3 BEGIN
4     UPDATE "TB_LIBRETA"
5     SET
6         str_nombre_libreta = VarNombreLibreta,
7         str_estado_libreta = VarEstadoLibreta
8     WHERE str_id_libreta = VarIdLibreta;
9 END;
10 /
Procedimiento creado.

SQL> set autotrace on;
SQL> exec "sp_ActualizarLibreta" ('Lib001','Calificaciones','Inactivo');
Procedimiento PL/SQL terminado correctamente.
```

```
STR_ID_LIBRETA          STR_NOMBRE_LIBRETA      STR_ESTADO
-----
Lib001                  Calificaciones          Inactivo

SQL>
```

Delete:

No podemos borrar ninguna libreta ya que la LOEI pide tener las calificaciones de los estudiantes en todo momento. Si se buscase realizar esto se debería tener una autorización legal porque el registro de notas es parte de la Base Legal.

tb_ubicacion	
pk,fk	int_index_persona
pk,fk	str_id_materia
pk,fk	str_id_aula
pk,fk	str_id_institucion
	str_horario

tb_ubicacion

int_index_persona
str_id_materia
str_id_aula
str_id_institucion
str_horario



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
SISTEMAS DE BASES DE DATOS

¿Que almacena la tabla Ubicación?

La entidad ubicacion nace de la relación que existe entre aula y materia almacenando las ids de ambas, donde también se encontrará la id del docente que imparte dicha materia, también se encuentra la id de la institución donde se encuentra el aula y el respectivo horario donde se imparte la materia

Insert:

Podemos crear nuevos registros porque de esta forma podemos crear los cronogramas de acuerdo al año escolar siendo porque por lo general por cada año escolar se generan nuevos cronogramas.

```
SQL> create or replace procedure "sp_CrearUbicacion" (int_index_persona NUMBER, str_id_institucion VARCHAR2,  
2 str_id_aula VARCHAR2, str_id_materia VARCHAR2, str_horario VARCHAR2)  
3 AS  
4 Begin  
5     INSERT INTO "TB_UBICACION" VALUES(int_index_persona, str_id_institucion, str_id_aula, str_id_materia, str_horario);  
6 End;  
7 /
```

Procedimiento creado.

```
SQL> exec "sp_CrearUbicacion" (1, 'Ins001', 'Au001', 'Mat001', '8:00-10:00 L-M-V');
```

Procedimiento PL/SQL terminado correctamente.

```
SQL> Select * from tb_ubicacion;
```

INT_INDEX_PERSONA	STR_ID_INS	STR_ID_AULA	STR_ID_MATERIA
1	Ins001	Au001	Mat001
8:00-10:00 L-M-V			

Update:

Podemos actualizar el horario, ya que puede ser en distintas jornadas o modalidades, también un horario en donde el docente puede estar disponible.

```
SQL> CREATE OR REPLACE PROCEDURE "sp_ActualizarUbicacion" (int_index_persona NUMBER, str_id_institucion VARCHAR2,  
2 str_id_aula VARCHAR2, str_id_materia VARCHAR2, Varhorario VARCHAR2)  
3 AS  
4 BEGIN  
5     UPDATE "TB_UBICACION"  
6     SET  
7     STR_HORARIO=VarHorario  
8     WHERE INT_INDEX_PERSONA = int_index_persona AND STR_ID_INSTITUCION = str_id_institucion  
9     AND STR_ID_AULA = str_id_aula AND STR_ID_MATERIA = str_id_materia;  
10 END;  
11 /
```

Procedimiento creado.



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
SISTEMAS DE BASES DE DATOS

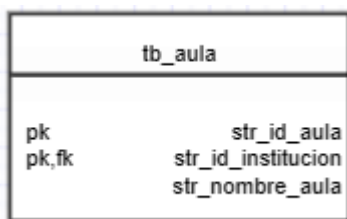
```
SQL> exec "sp_ActualizarUbicacion" (1,'Ins001','Au001','Mat001','7:00-9:00 L-M-V');  
  
Procedimiento PL/SQL terminado correctamente.  
  
SQL> Select * from tb_ubicacion;  
  
INT_INDEX_PERSONA STR_ID_INS STR_ID_AULA STR_ID_MATERIA  
-----  
STR_HORARI  
-----  
1 Ins001 Au001 Mat001  
7:00-9:00  
L-M-V
```

Delete:

Podemos borrar algún registro de la tabla ya que pueden darse casos donde la materia se deja de impartir o el aula deja de existir y como no afecta a otra tabla se la puede eliminar.

```
SQL> CREATE OR REPLACE PROCEDURE "sp_BorrarUbicacion" (int_index_persona NUMBER, str_id_institucion VARCHAR2,  
2 str_id_aula VARCHAR2, str_id_materia VARCHAR2)  
3 AS  
4 BEGIN  
5 DELETE FROM TB_UBICACION  
6 WHERE INT_INDEX_PERSONA = int_index_persona AND STR_ID_INSTITUCION = str_id_institucion  
7 AND STR_ID_AULA = str_id_aula AND STR_ID_MATERIA = str_id_materia;  
8 END;  
9 /  
  
Procedimiento creado.
```

```
SQL> exec "sp_BorrarUbicacion" (1,'Ins001','Au001','Mat001');  
  
Procedimiento PL/SQL terminado correctamente.  
  
SQL> Select * from tb_ubicacion;  
  
ninguna fila seleccionada
```



tb_aula
str_id_aula
str_id_institucion
str_nombre_aula

¿Que almacena la tabla Aula?

Lo que almacena la entidad aula son los campos que puede contener un aula dentro de una institución educativa y los campos que tenemos son los siguientes: el identificador de aula y de la institución y un nombre del aula, el id de la institución es un atributo heredado de la tabla Institución.

Insert:



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

SISTEMAS DE BASES DE DATOS

Podemos crear un registro en caso de que se construyese o se crease una nueva aula en la institución.

```
SQL> create or replace procedure "sp_CrearAula" (str_id_aula in varchar2, str_nombre_aula in varchar2, str_id_institucion varchar2)
2 AS
3 Begin
4     INSERT INTO "TB_AULA" VALUES(str_id_aula, str_nombre_aula, str_id_institucion);
5 End;
6 /

Procedimiento creado.

SQL> exec "sp_CrearAula" ('Au001','9no A','Ins001');

Procedimiento PL/SQL terminado correctamente.
```

STR_ID_AULA	STR_NOMBRE_AULA	STR_ID_INS
Au001	9no A	Ins001

SQL> |

Update:

Se puede actualizar solamente el nombre del aula porque el aula puede ser un año como ejemplo 8vo-C y al siguiente 9no-A. Sin embargo la id no cambia porque se debe conocer el aula al que nos referimos mediante la id.

```
SQL> CREATE OR REPLACE PROCEDURE "sp_ActualizarAula" (VarIdAula in varchar2, VarNombreAula in varchar2, VarIdIns in varchar2)
2 AS
3 BEGIN
4     UPDATE "TB_AULA"
5     SET
6     STR_NOMBRE_AULA= VarNombreAula,
7     STR_ID_INSTITUCION = VarIdInsS
8     WHERE STR_ID_AULA = VarIdAula;
9 END;
10 /

Advertencia: Procedimiento creado con errores de compilaci n.

SQL> set autoprint on;
SQL> exec "sp_ActualizarAula" ('Au001','10mo A','Ins001');
```

```
SQL> SELECT * FROM TB_AULA;
```

STR_ID_AULA	STR_NOMBRE_AULA	STR_ID_INS
Au001	10mo A	Ins001

SQL> |

Delete:

No se puede borrar el registro ya que es parte de una Institución y los únicos casos que puede pasar esto es si la Institución deja de funcionar o la edificación deja de existir. En caso de que se quiera dejar de usar el aula se puede cambiar el nombre a Libre y así tener el registro de que ese aula ya no se usa para impartir alguna materia.



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
SISTEMAS DE BASES DE DATOS

tb_institucion	
pk	str_id_institucion
fk	str_id_base
	str_nombre_institucion
	str_modalidad_institucion
	str_jornada_institucion

tb_institucion
str_id_institucion
str_id_base
str_nombre_institucion
str_modalidad_institucion
str_jornada_institucion

¿Que almacena la tabla Institución?

En la tabla institución almacenamos los campos referentes a la id, el nombre, la modalidad y la jornada, mencionando que también se debe tener el identificador de la base legal con la cual está trabajando y comprobar que cumple la LOEI.

Insert:

Se puede agregar una nueva institución en caso de que se lo necesite ya que siempre puede darse el caso de expansión.

```
SQL> create or replace procedure "sp_CrearInstitucion" (str_id_institucion in varchar2, str_nombre_institucion in varchar2,
2 str_modalidad_institucion in varchar2, str_jornada_institucion varchar2, str_id_base varchar2)
3 AS
4 Begin
5 INSERT INTO "TB_INSTITUCION" VALUES(str_id_institucion, str_nombre_institucion, str_modalidad_institucion, str_jornada_institucion, str_id_base);
6 End;
7 /

Procedimiento creado.

SQL> exec "sp_CrearInstitucion" ('Ins001','ESPE','Presencial','Matutina','BL001');

Procedimiento PL/SQL terminado correctamente.
```

STR_ID_INSTITUCION	STR_NOMBRE_INSTITUCI	STR_MODALI	STR_JORNAD	STR_ID_BAS
Ins001	ESPE	Presencial	Matutina	BL001

SQL> |

Update:

Se pueden actualizar solo los datos de la modalidad o jornada de la institución si así lo considera la institución, sin embargo los otros campos no se puede porque la institución se debe encontrar registrada en la AMIE como lo dicta la LOEI.

```
SQL> CREATE OR REPLACE PROCEDURE "sp_ActualizarInstitucion" (VarIdIns in varchar2, VarNombreIns in varchar2,
2 VarModalidad in varchar2, VarJornada varchar2, VarIdBase varchar2)
3 AS
4 BEGIN
5 UPDATE "TB_INSTITUCION"
6 SET
7 str_nombre_institucion= VarNombreIns,
8 str_modalidad_institucion = VarModalidad,
9 str_jornada_institucion = VarJornada,
10 str_id_base = VarIdBase
11 WHERE str_id_institucion = VarIdIns;
12 END;
13 /

Procedimiento creado.

SQL> set autoprint on;
SQL> exec "sp_ActualizarInstitucion" ('Ins001','ESPE','Online','Vespertina','BL001');

Procedimiento PL/SQL terminado correctamente.
```



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
SISTEMAS DE BASES DE DATOS

```
SQL> SELECT * FROM TB_INSTITUCION;
```

STR_ID_INSTITUCION	STR_NOMBRE_INSTITUCI	STR_MODALI	STR_JORNAD	STR_ID_BAS
Ins001	ESPE	Online	Vespertina	BL001

```
SQL> |
```

Delete:

No podemos borrar ningún registro de la institución ya que existen tablas vinculadas a la misma y los únicos casos que se podrían serian si no se lo encuentra vinculada en algún otro registro.

tb_base_legal	
pk	str_id_base str_nombre_base str_estado_base

tb_base_legal
str_id_base
str_nombre_base
str_estado_base

¿Que almacena la tabla Base Legal?

En la base legal almacenamos cada artículo que conste en la ley para que la institución educativa funcione de manera correcta y en esta entidad hemos declarado los siguientes campos, que son el identificador, el nombre y estado de la base legal.

Insert:

No podemos crear ninguna otra ley dentro de la base legal que está declarada para educación, por ende no podemos crear ningún registro con los campos determinados porque ya están establecidos.

```
SQL> create or replace procedure "sp_CrearBaseLegal" (str_id_base in varchar2, str_nombre_base in varchar2, str_estado_base in varchar2)
2 AS
3 Begin
4     INSERT INTO "TB_BASE_LEGAL" VALUES(str_id_base, str_nombre_base, str_estado_base);
5 End;
6 /

Procedimiento creado.

SQL> exec "sp_CrearBaseLegal" ('BL001','Legalidad','Activo');

Procedimiento PL/SQL terminado correctamente.
```

```
SQL> select * from tb_base_legal;
```

STR_ID_BASE	STR_NOMBRE_BASE	STR_ESTADO
BL001	Legalidad	Activo

Update:

No podemos actualizar ningún campo de nuestra base legal.

Delete:



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

SISTEMAS DE BASES DE DATOS

No podemos borrar ningún registro de la base legal.

tb_usuario	
pk	str_id_usuario
fk	int_index_persona
	str_nombre_usuario
	str_password_usuario

tb_usuario

str_id_usuario
int_index_persona
str_nombre_usuario
str_password_usuario

¿Que almacena la tabla Usuario?

Lo que almacena la tabla usuario son las credenciales de cada persona que se registre, está compuesta con los siguientes campos, los identificadores del usuario y de la persona que hereda en usuario y como es un usuario va a tener siempre un nombre y contraseña.

Insert:

Si podemos crear un usuario porque por cada nuevo estudiante o docente se debe tener su respectivo usuario.

```
SQL> create or replace procedure "sp_CrearUsuario" (str_id_usuario VARCHAR2, str_nombre_usuario VARCHAR2, str_password_usuario VARCHAR2,
2 int_index_persona NUMBER)
3 AS
4 Begin
5 INSERT INTO TB_USUARIO VALUES(str_id_usuario, str_nombre_usuario, str_password_usuario, int_index_persona);
6 End;
7 /
Procedimiento creado.
```

```
SQL> exec "sp_CrearUsuario" ('User001', 'JCaicedo', '26092000', 1);
Procedimiento PL/SQL terminado correctamente.
```

```
SQL> select * from tb_usuario;

STR_ID_USUARIO      STR_NOMBRE_USUARIO  STR_PASSWORD_USUARIO INT_INDEX_PERSONA
-----
User001             JCaicedo            26092000             1
```

Update:

Podemos actualizar, pero solamente la contraseña puesto que los otros campos ya están definidos y no se pueden tocar.

```
SQL> CREATE OR REPLACE PROCEDURE "sp_ActualizarUsuario" (str_id_usuario VARCHAR2, VarPasswordUsuario VARCHAR2)
2 AS
3 BEGIN
4 UPDATE "TB_USUARIO"
5 SET
6 STR_PASSWORD_USUARIO=VarPasswordUsuario
7 WHERE STR_ID_USUARIO = str_id_usuario;
8 END;
9 /
Procedimiento creado.
```



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

SISTEMAS DE BASES DE DATOS

```
SQL> exec "sp_ActualizarUsuario" ('User001','jg260900');  
  
Procedimiento PL/SQL terminado correctamente.  
  
SQL> select * from tb_usuario;  
  
STR_ID_USUARIO      STR_NOMBRE_USUARIO      STR_PASSWORD_USUARIO      INT_INDEX_PERSONA  
-----  
User001              JCaicedo                  jg260900                    1
```

Delete:

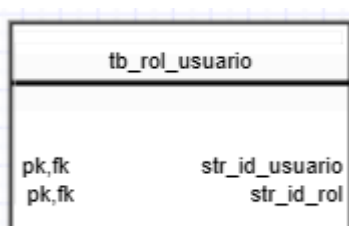
Podemos borrar al usuario si ya no pertenece a la institución o no se encuentra matriculado durante más de 2 años, en ese caso se eliminaría todo el registro, caso contrario no se puede eliminar el registro.

```
SQL> CREATE OR REPLACE PROCEDURE "sp_BorrarUsuario" (str_id_usuario VARCHAR2)  
2 AS  
3 BEGIN  
4     DELETE FROM TB_USUARIO  
5     WHERE STR_ID_USUARIO = str_id_usuario;  
6 END;  
7 /
```

Procedimiento creado.

```
SQL> exec "sp_BorrarUsuario" ('User001');  
  
Procedimiento PL/SQL terminado correctamente.
```

```
SQL> select * from tb_usuario;  
  
ninguna fila seleccionada
```



tb_rol_usuario

str_id_usuario

str_id_rol

¿Que almacena la tabla Rol_Usuario?

Esta tabla se crea por la relación que tiene entre rol y el usuario, en esta tabla pasan a heredar los identificadores de usuario y rol para identificar los roles de los usuarios.

Insert:

No podemos crear ningún registro en esta entidad porque los identificadores ya están definidos, si se buscase realizar esto se debería ser un usuario administrador o un usuario con los permisos respectivos para esta acción.

Update:



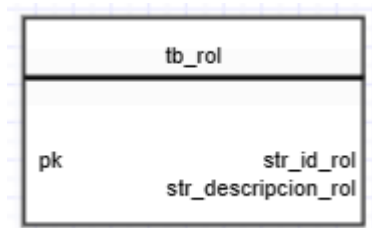
UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

SISTEMAS DE BASES DE DATOS

No podemos actualizar ningún campo de esta entidad, porque al actualizar un identificador se producen errores al seguir las siguientes relaciones en la tablas.

Delete:

No podemos borrar ningún registro de esta tabla, si se buscase realizar esto se debería ser un usuario administrador o un usuario con los permisos respectivos para esta acción..



tb_rol
str_id_rol
str_descripcion_rol

¿Que almacena la tabla Rol?

La tabla rol almacena todos los roles que pueden tener los usuarios dentro de la institución educativa, por ejemplo como administrador.

Insert:

No podemos crear roles porque son partes de la base de datos, si se buscase realizar esto se debería ser un usuario administrador o un usuario con los permisos respectivos para esta acción.

```
SQL> create or replace procedure "sp_CrearRol" (str_id_rol in varchar2, str_descripcion_rol in varchar2)
2 AS
3 Begin
4     INSERT INTO "TB_ROL" VALUES(str_id_rol, str_descripcion_rol);
5 End;
6 /

Procedimiento creado.
```

```
SQL> exec "sp_CrearRol" ('Rol003','Estudiante');

Procedimiento PL/SQL terminado correctamente.
```

Update:

No podemos actualizar los roles porque son partes de la base de datos, si se buscase realizar esto se debería ser un usuario administrador o un usuario con los permisos respectivos para esta acción.

```
SQL> CREATE OR REPLACE PROCEDURE "sp_ActualizarRol" (VarIdRol in varchar2, VarDescripcionRol in varchar2)
2 AS
3 BEGIN
4     UPDATE "TB_ROL"
5     SET
6     STR_DESCRIPCION_ROL = VarDescripcionRol
7     WHERE STR_ID_ROL = VarIdRol;
8 END;
9 /

Procedimiento creado.
```



UNIVERSIDAD DE LAS FUERZAS ARMADAS" ESPE"
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
SISTEMAS DE BASES DE DATOS

```
SQL> exec "sp_ActualizarRol" ('Rol003','Administrador');  
  
Procedimiento PL/SQL terminado correctamente.
```

STR_ID_ROL	STR_DESCRIPCION_ROL
Rol001	Administrador
Rol002	Estudiante
Rol003	Administrador

```
SQL>
```

Delete:

No podemos borrar porque son partes de la base de datos, si se buscase realizar esto se debería ser un usuario administrador o un usuario con los permisos respectivos para esta acción.

Como parte del documento tambien se posee a disposición un enlace a GitHub donde se encuentra toda la documentación del codigo y capturas de los procedimientos realizados. En GitHub tambien se podra encontrar el script creado para la realizacion de la presente tarea. Para poder acceder al codigo puede ingresar a la pagina web:

[jgcaicedo1/-NRC_8393-_Tarea1SPF0Unidad3_CaicedoViteri_JorgeGabriel_Quishpe_Guaytarilla_AnthonyEsteven \(github.com\)](https://github.com/jgcaicedo1/-NRC_8393-_Tarea1SPF0Unidad3_CaicedoViteri_JorgeGabriel_Quishpe_Guaytarilla_AnthonyEsteven)