

En este apunte se verán las sentencias INSERT, UPDATE y DELETE.

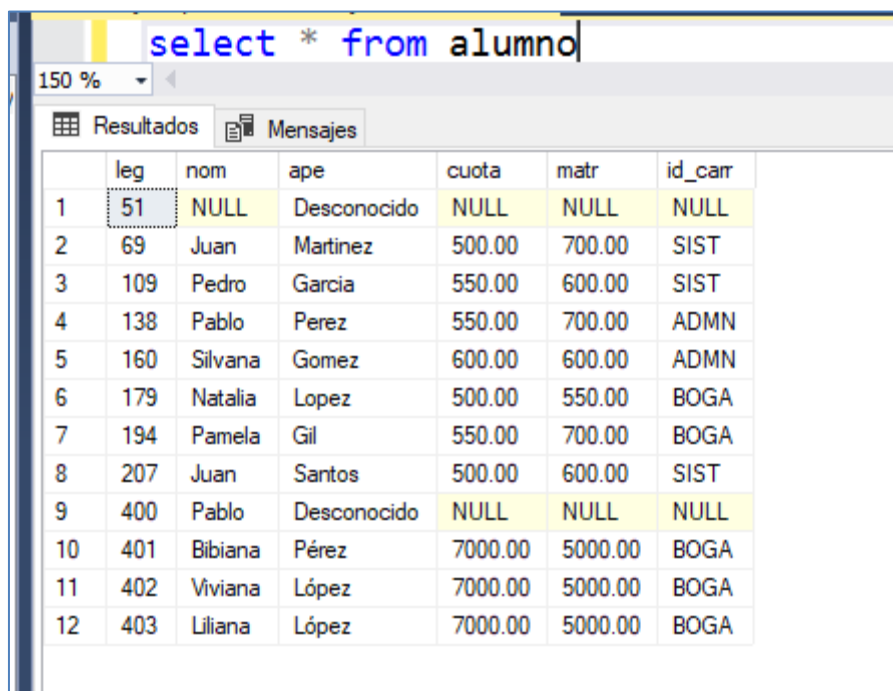
INSERT: permite ingresar registros a una tabla.

INSERT DE UNA FILA:

Sintaxis:

- INSERT INTO tabla (columna1, columna2,.....,columnaN)
VALUES (valor1, valor2,.....,valorN)

EJEMPLO 40



```
select * from alumno
```

	leg	nom	ape	cuota	matr	id_carr
1	51	NULL	Desconocido	NULL	NULL	NULL
2	69	Juan	Martinez	500.00	700.00	SIST
3	109	Pedro	Garcia	550.00	600.00	SIST
4	138	Pablo	Perez	550.00	700.00	ADMN
5	160	Silvana	Gomez	600.00	600.00	ADMN
6	179	Natalia	Lopez	500.00	550.00	BOGA
7	194	Pamela	Gil	550.00	700.00	BOGA
8	207	Juan	Santos	500.00	600.00	SIST
9	400	Pablo	Desconocido	NULL	NULL	NULL
10	401	Bibiana	Pérez	7000.00	5000.00	BOGA
11	402	Viviana	López	7000.00	5000.00	BOGA
12	403	Liliana	López	7000.00	5000.00	BOGA

Insertar un registro en la tabla ALUMNO con los siguientes datos:

Legajo52: nombre: Jorge, apellido: García, cuota, 5000, matrícula: 7000 y carrera: SIST

INSTRUCCIÓN AGREGADA

```
INSERT INTO alumno  
values (52, 'Jorge', 'García', 5000, 7000, 'SIST');
```

NOTA: los valores alfanuméricos deben guardarse entre comillas, los valores numéricos sin comillas y los valores con tipo fecha depende del SGBD, por ejemplo en SQL Server se guardan entre comillas simples.

	leg	nom	ape	cuota	matr	id_carr
1	51	NULL	Desconocido	NULL	NULL	NULL
2	52	Jorge	García	5000.00	7000.00	SIST
3	69	Juan	Martinez	500.00	700.00	SIST
4	109	Pedro	Garcia	550.00	600.00	SIST
5	138	Pablo	Perez	550.00	700.00	ADMN
6	160	Silvana	Gomez	600.00	600.00	ADMN
7	179	Natalia	Lopez	500.00	550.00	BOGA
8	194	Pamela	Gil	550.00	700.00	BOGA
9	207	Juan	Santos	500.00	600.00	SIST
10	400	Pablo	Desconocido	NULL	NULL	NULL
11	401	Bibiana	Pérez	7000.00	5000.00	BOGA
12	402	Viviana	López	7000.00	5000.00	BOGA
13	403	Liliana	López	7000.00	5000.00	BOGA

En el caso de que la columna admita NULL, puede omitirse en el INSERT y el SGBD lo completará automáticamente.

EJEMPLO 41

Insertar los siguientes datos del alumno con legajo 150:

Nombre: Javier, apellido: Bermudez, código de carrera: SIST

INSTRUCCIÓN AGREGADA

```
INSERT INTO alumno (leg, nom, ape, id_carr)  
values (150, 'Javier', 'Bermudez', 'SIST');
```

Así queda la tabla nueva:

	leg	nom	ape	cuota	matr	id_carr
1	51	NULL	Desconocido	NULL	NULL	NULL
2	52	Jorge	García	5000.00	7000.00	SIST
3	69	Juan	Martinez	500.00	700.00	SIST
4	109	Pedro	Garcia	550.00	600.00	SIST
5	138	Pablo	Perez	550.00	700.00	ADMN
6	150	Javier	Bermudez	NULL	NULL	SIST
7	160	Silvana	Gomez	600.00	600.00	ADMN
8	179	Natalia	Lopez	500.00	550.00	BOGA
9	194	Pamela	Gil	550.00	700.00	BOGA
10	207	Juan	Santos	500.00	600.00	SIST
11	400	Pablo	Desconocido	NULL	NULL	NULL
12	401	Bibiana	Pérez	7000.00	5000.00	BOGA
13	402	Viviana	López	7000.00	5000.00	BOGA
14	403	Liliana	López	7000.00	5000.00	BOGA

EJEMPLO 42

Ingresar los datos del alumno con legajo **179**, nombre: Mariana, Apellido: Fernández Rojas.

```
INSERT INTO alumno (leg, nom, ape)
values (179, 'Mariana', 'Fernández Rojas');
```

Mensajes

Mens. 2627, Nivel 14, Estado 1, Línea 1
 Infracción de la restricción PRIMARY KEY 'PK_alumno_DE107A85E72EEA91'.
 No se puede insertar una clave duplicada en el objeto 'dbo.alumno'.
 El valor de la clave duplicada es (179).
 Se terminó la instrucción.

Como se ve en el ejemplo, el SGBD impide que se cargue este registro por integridad de la clave primaria, es decir que controla que NO haya 2 registros con la misma clave primaria, en este caso **leg** es clave primaria e impide que se carguen 2 registros con el legajo 179.

INSERT MULTIFILA

EJEMPLO 43

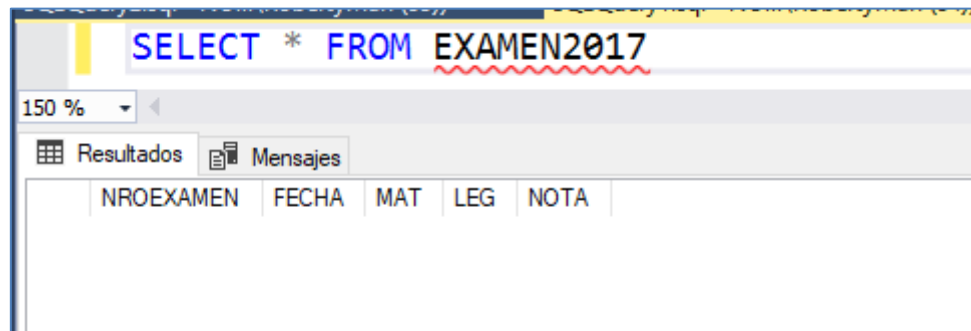
Dada la tabla examen2017 (nroexamen, fecha, mat, leg, nota) donde:

- El campo nroexamen tiene el mismo tipo de datos que nroexamen de la tabla examen
- El campo fecha tiene el mismo tipo de dato que el campo fecha de la tabla examen

- El campo mat tiene el mismo tipo de dato que el campo mat de la tabla examen
- El campo leg tiene el mismo tipo de dato que el campo leg de la tabla examen
- El campo nota tiene el mismo tipo de dato que el campo nota de la tabla examen

Ingresar en la tabla EXAMEN2017 todos los exámenes de la tabla EXAMEN que correspondan al año 2017 (Incluir todas las columnas).

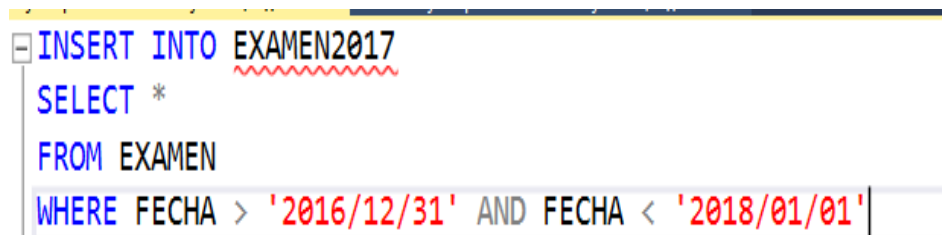
Veamos ahora la tabla EXAMEN2017 está vacía (como se muestra a continuación):



The screenshot shows a SQL query window with the query `SELECT * FROM EXAMEN2017`. Below the query, there are tabs for 'Resultados' and 'Mensajes'. The 'Resultados' tab is active, displaying a table with the following columns: NROEXAMEN, FECHA, MAT, LEG, and NOTA. The table is currently empty.

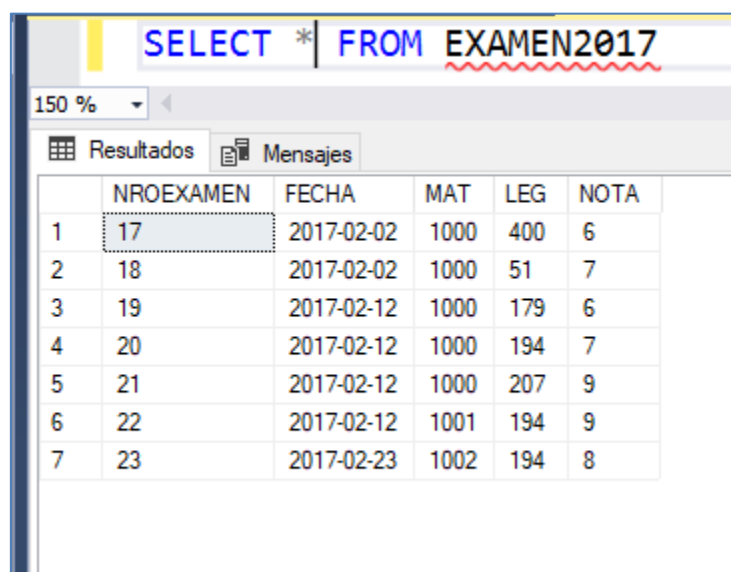
	NROEXAMEN	FECHA	MAT	LEG	NOTA
--	-----------	-------	-----	-----	------

Al ejecutar la sentencia de carga de datos:



The screenshot shows a SQL query window with the following query: `INSERT INTO EXAMEN2017 SELECT * FROM EXAMEN WHERE FECHA > '2016/12/31' AND FECHA < '2018/01/01'`. The query is highlighted in blue.

Obtendremos la tabla deseada, la cual podemos corroborar haciendo la siguiente consulta:



The screenshot shows a SQL query window with the query `SELECT * FROM EXAMEN2017`. Below the query, there are tabs for 'Resultados' and 'Mensajes'. The 'Resultados' tab is active, displaying a table with the following columns: NROEXAMEN, FECHA, MAT, LEG, and NOTA. The table contains 7 rows of data.

	NROEXAMEN	FECHA	MAT	LEG	NOTA
1	17	2017-02-02	1000	400	6
2	18	2017-02-02	1000	51	7
3	19	2017-02-12	1000	179	6
4	20	2017-02-12	1000	194	7
5	21	2017-02-12	1000	207	9
6	22	2017-02-12	1001	194	9
7	23	2017-02-23	1002	194	8

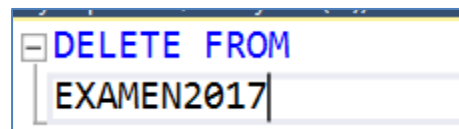
Delete: esta sentencia elimina filas de una tabla:

Sintaxis:

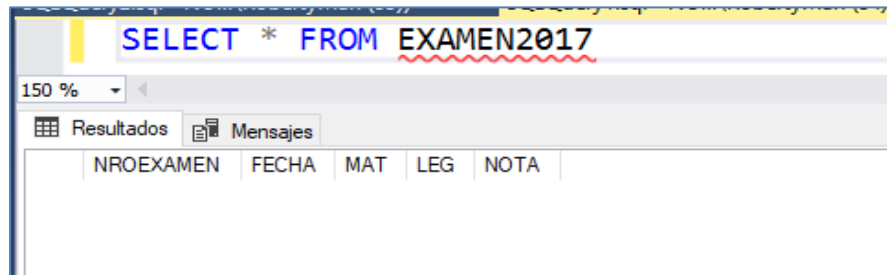
- DELETE FROM tabla: elimina todas las filas de una tabla.
- DELETE FROM tabla: elimina solo las filas que cumplan la condición específica WHERE condición

EJEMPLO 44

Eliminar todos los registros de la tabla EXAMEN2017.



Si revisamos ahora el contenido de la tabla EXAMEN2017, corroboramos que está vacía:



EJEMPLO 45

Dar de baja al alumno con legajo 52.

Veamos primero la tabla actual:

	leg	nom	ape	cuota	matr	id_carr
1	51	NULL	Desconocido	NULL	NULL	NULL
2	52	Jorge	García	5000.00	7000.00	SIST
3	69	Juan	Martinez	500.00	700.00	SIST
4	109	Pedro	Garcia	550.00	600.00	SIST
5	138	Pablo	Perez	550.00	700.00	ADMN
6	150	Javier	Bermudez	NULL	NULL	SIST
7	160	Silvana	Gomez	600.00	600.00	ADMN
8	179	Natalia	Lopez	500.00	550.00	BOGA
9	194	Pamela	Gil	550.00	700.00	BOGA
10	207	Juan	Santos	500.00	600.00	SIST
11	400	Pablo	Desconocido	NULL	NULL	NULL
12	401	Bibiana	Pérez	7000.00	5000.00	BOGA
13	402	Viviana	López	7000.00	5000.00	BOGA
14	403	Liliana	López	7000.00	5000.00	BOGA

Si ejecutamos la sentencia:

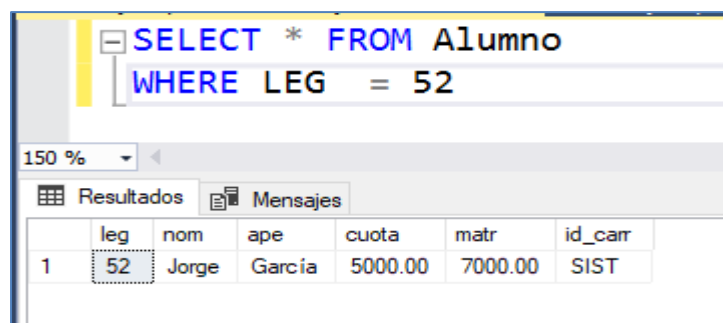
```
DELETE FROM ALUMNO
```

```
WHERE LEG = 52
```

Obtendremos la siguiente tabla:

	leg	nom	ape	cuota	matr	id_carr
1	51	NULL	Desconocido	NULL	NULL	NULL
2	69	Juan	Martinez	500.00	700.00	SIST
3	109	Pedro	Garcia	550.00	600.00	SIST
4	138	Pablo	Perez	550.00	700.00	ADMN
5	150	Javier	Bermudez	NULL	NULL	SIST
6	160	Silvana	Gomez	600.00	600.00	ADMN
7	179	Natalia	Lopez	500.00	550.00	BOGA
8	194	Pamela	Gil	550.00	700.00	BOGA
9	207	Juan	Santos	500.00	600.00	SIST
10	400	Pablo	Desconocido	NULL	NULL	NULL
11	401	Bibiana	Pérez	7000.00	5000.00	BOGA
12	402	Viviana	López	7000.00	5000.00	BOGA
13	403	Liliana	López	7000.00	5000.00	BOGA

NOTA: en la práctica es aconsejable ver los datos que se quieren modificar o eliminar antes de ejecutar la consulta, para no cometer errores graves, por ejemplo en el caso anterior se podría haber hecho previamente a la eliminación:



The screenshot shows a database query window with the following SQL statement:

```
SELECT * FROM Alumno  
WHERE LEG = 52
```

Below the query, there are tabs for 'Resultados' (Results) and 'Mensajes' (Messages). The 'Resultados' tab is active, displaying a table with the following data:

	leg	nom	ape	cuota	matr	id_carr
1	52	Jorge	García	5000.00	7000.00	SIST

Para estar seguro que solo se eliminará ese registro y, luego sí, ejecutar la sentencia:
DELETE

EJEMPLO 46

Dar de baja a los cursos del profesor José Pérez.

Nota: en este caso, el registro que necesitamos eliminar está en la tabla CURSO, pero para saber cuáles son los cursos del profesor que se llama José Pérez necesitamos la tabla PROFESOR. El ANSI-ISO de SQL establece que solo se puede usar una tabla en el FROM, por lo tanto si fuese necesario juntar los datos de esta tabla con datos de otras tablas, deberá hacerse mediante la cláusula SELECT.

Por lo tanto, la sentencia a ejecutar sería:

```
DELETE from CURSO
WHERE prof in (select prof
FROM PROFESOR WHERE nya like 'Jose Perez')
```

150 %

Mensajes

(2 filas afectadas)

EJEMPLO 47

Veamos qué sucede si queremos eliminar los datos de algún alumno que tiene exámenes rendidos, como es el caso del alumno con legajo 51 (por ejemplo, el 2 de febrero de 2017 rindió un examen). Acorde a cómo se haya definido la clave foránea en el catálogo, el SGBD ejecutará la sentencia o lo impedirá. Como en este caso se definió que no permita la eliminación de alumnos con exámenes, el SGBD deberá cumplir la integridad referencial.

Veamos entonces qué ocurre:

```
DELETE FROM alumno
WHERE LEG = 51
```

150 %

Mensajes

Mens. 547, Nivel 16, Estado 0, Línea 1
Instrucción DELETE en conflicto con la restricción FOREIGN KEY 'FK_final_leg_656C112C'.
El conflicto ha aparecido en la base de datos 'FACULTAD', tabla 'dbo.alumno', column 'leg'.
Se terminó la instrucción.

EJEMPLO 48

Lo mismo ocurre en el siguiente caso:

Dar de baja al alumno que rindió el examen número 14.

En este caso, el registro que tenemos que eliminar está en la tabla ALUMNO, pero para saber cuál es, necesitamos buscar el legajo en la tabla EXAMEN. Por consiguiente, además de la tabla ALUMNO, necesitamos la tabla EXAMEN. El ANSI-ISO de SQL establece que solo se puede usar una tabla en el FROM, por lo tanto si fuese necesario juntar los datos de esta tabla con datos de otras tablas, deberá hacerse mediante la cláusula SELECT.

Entonces, la sentencia a ejecutar sería:

```
DELETE FROM ALUMNO
WHERE LEG IN (SELECT LEG
              FROM EXAMEN WHERE nroexamen =14)
```

Mensajes

Mens. 547, Nivel 16, Estado 0, Línea 1
Instrucción DELETE en conflicto con la restricción FOREIGN KEY 'FK_final_leg_656C112C'.
El conflicto ha aparecido en la base de datos 'FACULTAD', tabla 'dbo.alumno', column 'leg'.
Se terminó la instrucción.

Nota: a pesar de que la sintaxis de la sentencia es correcta, nuevamente el SGBD impide que se borre un registro que corresponde a un alumno que tiene algún examen.

EJEMPLO 49

Dar de baja a los alumnos que NUNCA rindieron examen.

```
DELETE FROM ALUMNO
WHERE LEG not in (SELECT distinct LEG FROM EXAMEN)
```

Mensajes

(3 filas afectadas)

UPDATE: esta sentencia permite modificar los valores de los campos de uno o varios registros de una tabla.

SINTAXIS:

UPDATE tabla

SET campo1 = X1, campo2 = x2, campoN = xn

WHERE condición

Al igual que en el SELECT y DELETE, la cláusula WHERE es optativa.

Veamos primero nuestra tabla de alumnos para poder entender luego los cambios que produce esta sentencia.

select * FROM ALUMNO

150 %

Resultados Mensajes

	leg	nom	ape	cuota	matr	id_carr
1	51	NULL	Desconocido	NULL	NULL	NULL
2	69	Juan	Martinez	500.00	700.00	SIST
3	109	Pedro	Garcia	550.00	600.00	SIST
4	138	Pablo	Perez	550.00	700.00	ADMN
5	160	Silvana	Gomez	600.00	600.00	ADMN
6	179	Natalia	Lopez	500.00	550.00	BOGA
7	194	Pamela	Gil	550.00	700.00	BOGA
8	207	Juan	Santos	500.00	600.00	SIST
9	400	Pablo	Desconocido	NULL	NULL	NULL
10	401	Bibiana	Pérez	7000.00	5000.00	BOGA

EJEMPLO 50

Incrementar un 20% el valor de la matrícula y un 10% el valor de la cuota de todos los alumnos.

```
UPDATE alumno
SET matr = matr * 1.20 , cuota = cuota * 1.10
```

La tabla obtenida será:

	leg	nom	ape	cuota	matr	id_carr
1	51	NULL	Desconocido	NULL	NULL	NULL
2	69	Juan	Martinez	550.00	840.00	SIST
3	109	Pedro	Garcia	605.00	720.00	SIST
4	138	Pablo	Perez	605.00	840.00	ADMN
5	160	Silvana	Gomez	660.00	720.00	ADMN
6	179	Natalia	Lopez	550.00	660.00	BOGA
7	194	Pamela	Gil	605.00	840.00	BOGA
8	207	Juan	Santos	550.00	720.00	SIST
9	400	Pablo	Desconocido	NULL	NULL	NULL
10	401	Bibiana	Pérez	7700.00	6000.00	BOGA

EJEMPLO 51

Actualizar el precio de la matrícula de los alumnos de la carrera Abogacía, asignándole un valor de \$ 7000.

NOTA: se debe actualizar los registros de la tabla Alumno, pero el nombre de la carrera está guardado en la tabla CARRERA, entonces, al igual que con la sentencia DELETE, para cumplir con el estándar ANSI-ISO relacionaremos ambas tablas con la cláusula SELECT.

```

UPDATE alumno
SET matr = 7000
WHERE id_carr IN (SELECT id
                  FROM carrera
                  WHERE nom like 'ABOGACIA' )

```

	leg	nom	ape	cuota	matr	id_carr
1	51	NULL	Desconocido	NULL	NULL	NULL
2	69	Juan	Martinez	550.00	840.00	SIST
3	109	Pedro	Garcia	605.00	720.00	SIST
4	138	Pablo	Perez	605.00	840.00	ADMN
5	160	Silvana	Gomez	660.00	720.00	ADMN
6	179	Natalia	Lopez	550.00	7000.00	BOGA
7	194	Pamela	Gil	605.00	7000.00	BOGA
8	207	Juan	Santos	550.00	720.00	SIST
9	400	Pablo	Desconocido	NULL	NULL	NULL
10	401	Bibiana	Pérez	7700.00	7000.00	BOGA