

Proyecto final Base de Datos I

Sistema de Notas para un Colegio



Proyecto Final

Base de datos I

Ing. Andrea Belén De La Cruz Hernández

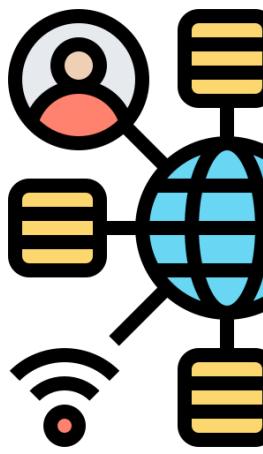
Integrantes del grupo:

1. Elvis Diego Josué Quiejú Damián 2290-18-8721
2. Rudy Roberto Méndez Galindo 2290-19-7357
3. Evelyn Cristina Roquel Cululén 2290-19-17485
4. William Denis Fernando Chiroy 22290-15-14801
5. Manuel Adrian Chet Samayoa 2290-20-15833

Sololá 22 de Octubre 2022

Contents

PROBLEMÁTICA	4
ANTECEDENTES	5
Diagrama Entidad Relación.....	6
Diagrama Relacional.....	7
Normalización.....	8
Programas utilizados.....	10
CONSULTAS	12
SUBCONSULTAS	24
JOINS	Error! Bookmark not defined.
PROCEDIMIENTOS ALMACENADOS	Error! Bookmark not defined.



Sistema de Notas Colegio

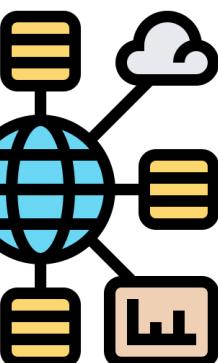
Como proyecto final de curso de base de datos I se solicito que se creara una base de datos para un sistema de entrega de notas de alumnos para un colegio, por lo tanto se planteó lo siguiente:

PROBLEMÁTICA

La población estudiantil aumentó durante la pandemia, por restricciones del área de salud no se podía realizar reuniones con padres de familia para entregar notas, por lo que se busca crear un sistema donde se facilite la entrega de notas a los alumnos del colegio y así facilitar procesos y agilizar la entrega de las mismas a los alumnos solventes.

SOLUCIÓN

Se implementará una base de datos donde se muestren las notas de los alumnos, para que de esta forma se pueda llevar un control de los punteos y calificaciones que tendrá cada alumno, el objetivo será facilitar el manejo y la entrega de notas.





ANTECEDENTES

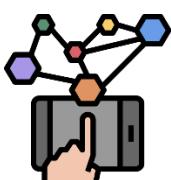
Para el desarrollo de la base de datos para el sistema de notas para un centro educativo, se evaluó el problema planteado, ya que se tomó de base el ingreso de la tecnología a los centros educativos; convencionalmente en los centros educativos de nivel primaria, básico y diversificado de Guatemala se realizan entrega de notas o calificaciones trimestralmente mediante una reunión donde los encargados o representantes de los alumnos inscritos en el establecimiento se presentaban a recibir una boleta de papel la cual contenía las calificaciones del alumno, este sistema fue funcional por muchos años sin embargo a partir del año 2020 a consecuencia de la pandemia causada por el covid-19 el sistema educativo tuvo que realizar cambios en sus métodos para impartir clases, entrega de reportes y notas de los alumnos.

A nivel mundial, el cierre de los establecimientos educativos para prevenir los contagios y con ello proteger a las niñas, niños y adolescentes fue una constante en los sistemas de educación. Esta realidad ha presentado efectos inmediatos como la pérdida de aprendizaje, lo que a su vez amenaza con deteriorar aún más los resultados educativos.

A raíz de este cambio tan radical en la vida estudiantil se toma la iniciativa por parte de algunos centros educativos en afiliarse a distintas plataformas para poder entregar notas, llevar un control de las tareas entregadas y los exámenes presentados, así mismo llevar un control donde el tutor, alumno y maestro puedan ver el avance del alumno en el trimestre. Facilitando así la entrega de los reportes y las entregas de notas sin necesidad de reunirse de manera presencial. Han pasado 2 años desde que la pandemia ingresó a Guatemala y muchos establecimientos han decidido seguir utilizando las plataformas y sistemas tecnológicos para llevar su control y agilizar sus procesos.

Tomando en cuenta esta información que se utiliza como base para el desarrollo de nuestra base de datos, teniendo como objetivo solucionar la problemática planteada y realizando un sistema funcional y fácil de entender para el usuario, así mismo que le permita ordenar y buscar los datos que necesita de manera fácil y rápido aprovechando de la mejor manera los recursos del sistema, se desarrolla un sistema generalizado que cumple con los requisitos básicos para ser utilizado.

Los sistemas han ingresado al sistema educativo para quedarse según el artículo 4 del reglamento de Centros Educativos, “*El Ministerio de Educación creará las herramientas electrónicas de aplicación que estarán a disposición de los interesados y publicados en su portal electrónico, a través del Módulo de Centros educativos privados del Sistema Nacional de Registros Educativos*”. Por tal motivo se deduce que implementar sistemas electrónicos en la educación será una de las mejores alternativas para facilitar el trabajo de los maestros y agilizar los procesos de los alumnos y autoridades educativas.



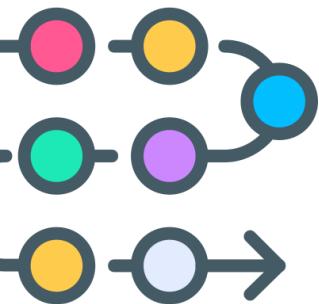


Diagrama Entidad Relación

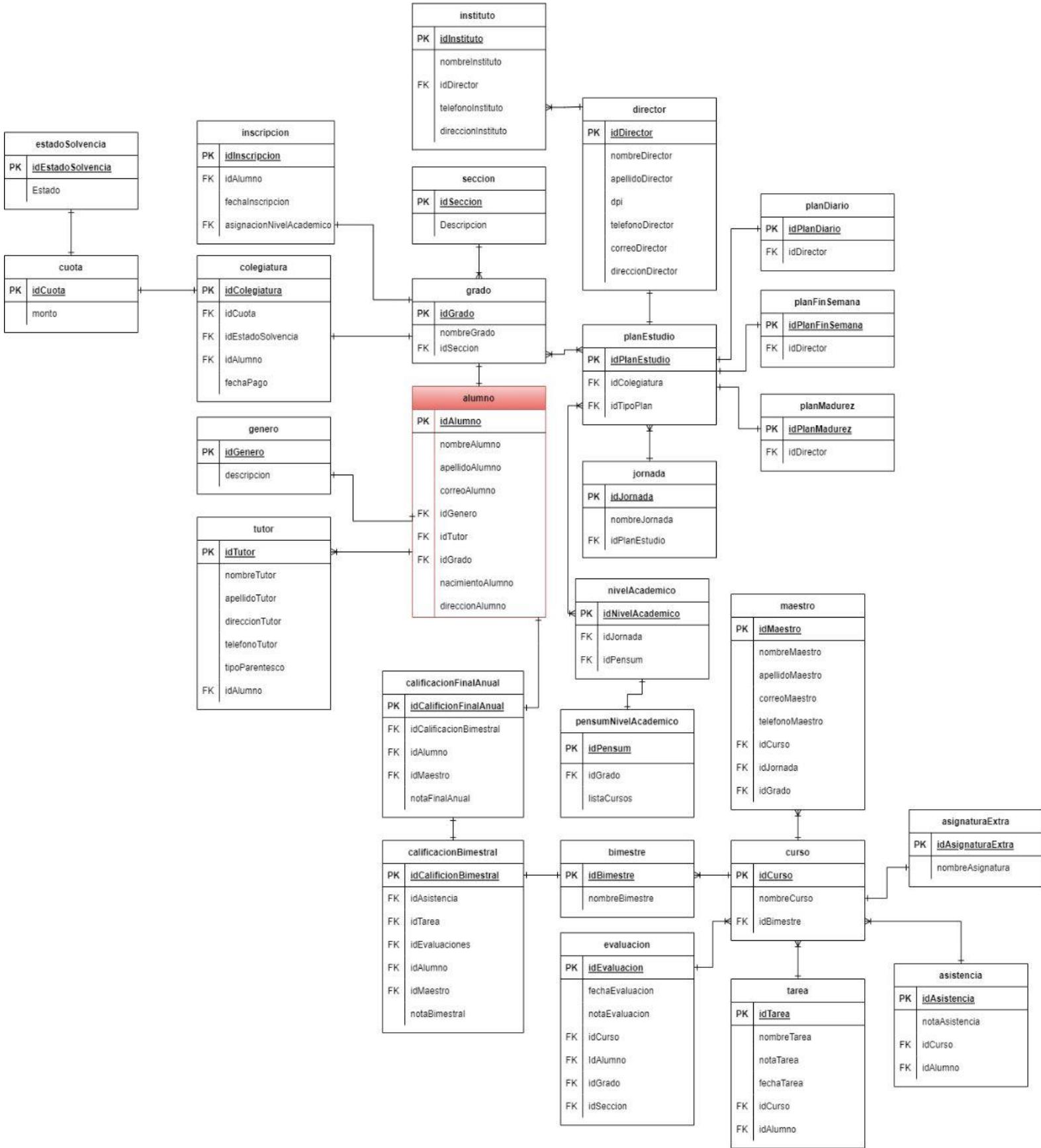
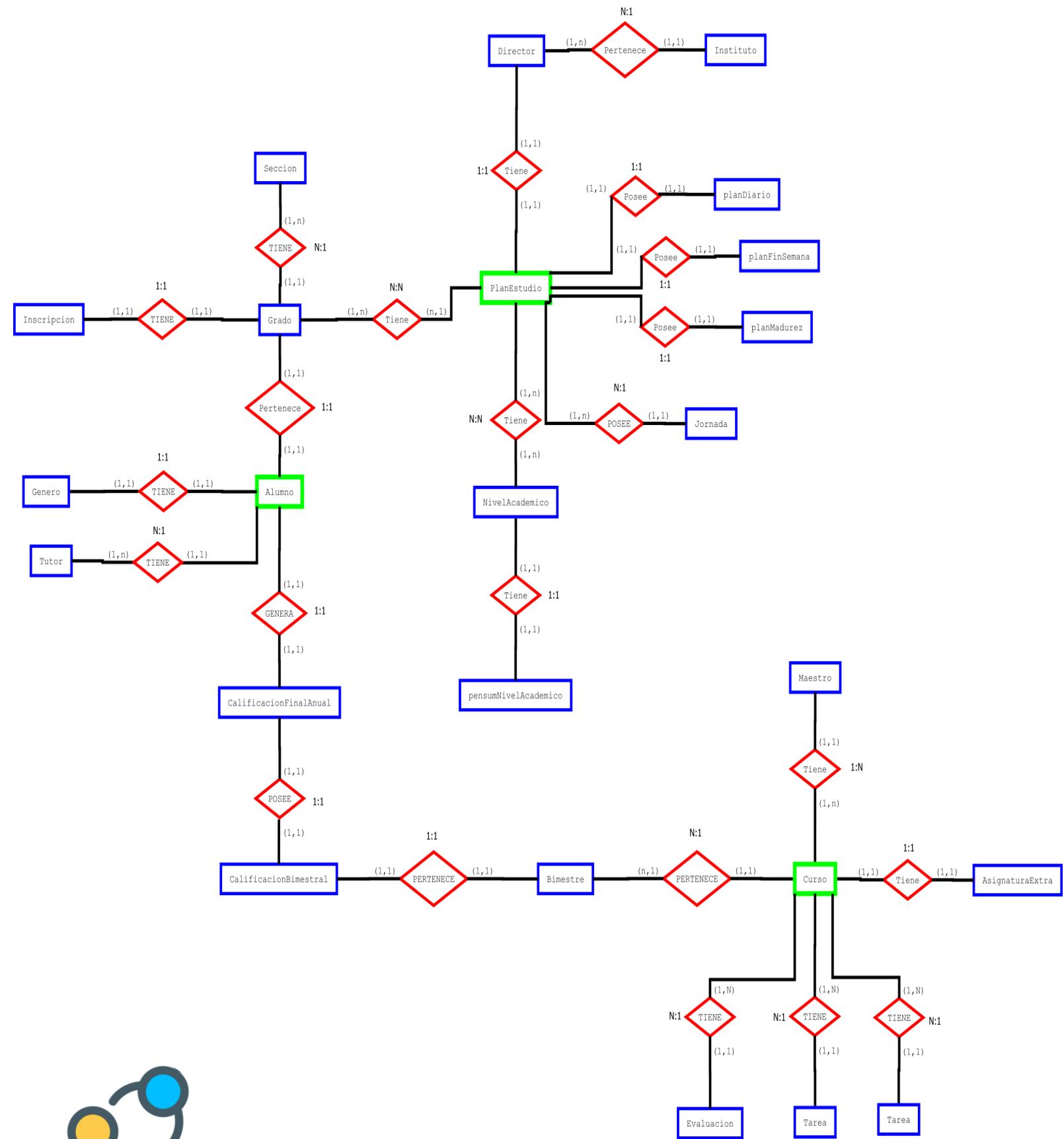


Diagrama Relacional



Normalización



Para el presente proyecto se aplico una normalización con el fin de organizar los datos del sistema, se realizaron los diagramas relacionales y los diagramas entidad relación, se incluyeron la creación de tablas y el establecimiento entre ellas según las reglas diseñadas.

Se buscaba que la base de datos sea flexible, funcional y eliminar la redundancia y las dependencias incoherentes ya que los datos redundantes desperdician espacio y en el futuro crearan problemas de mantenimiento ya que se encuentran iguales en más de un lugar.

Para poder realizar la normalización en el proyecto se utilizaron los tres pasos establecidos para normalizar los cuales fueron:

Primera Forma Normal: Se crearon las tablas y los diagramas relacionales y se eliminan los grupos en las tablas individuales, se identificaron los conjuntos de datos relacionados con una llave primaria.

Segunda Forma Normal: Se relacionaban las tablas y se crearon tablas independientes

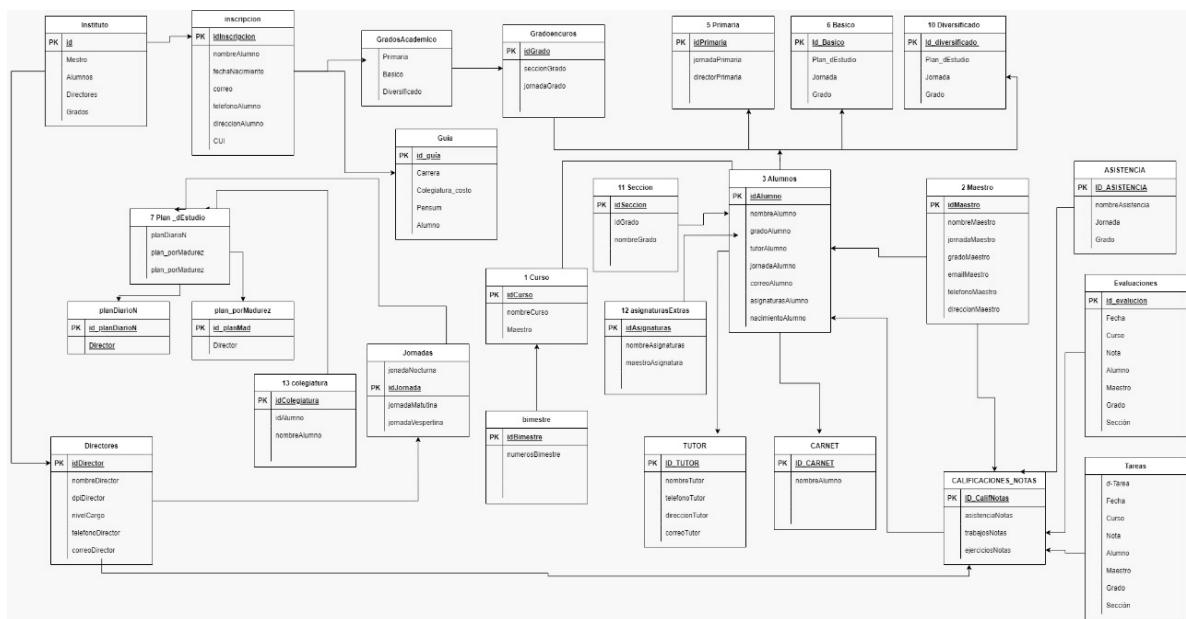
Tercera Forma Normal: Se eliminaron los campos y las tablas que no eran útiles dentro del sistema.

Para cerciorarnos de que las tablas estuvieran normalizadas en la base de datos se verifico que cumpliera con lo siguiente:

- Cada tabla tiene un nombre único.
- No se permitió que hubiera datos duplicados.
- Se aplicaron las reglas de normalización.
- Se verifico la eliminación de redundancias.

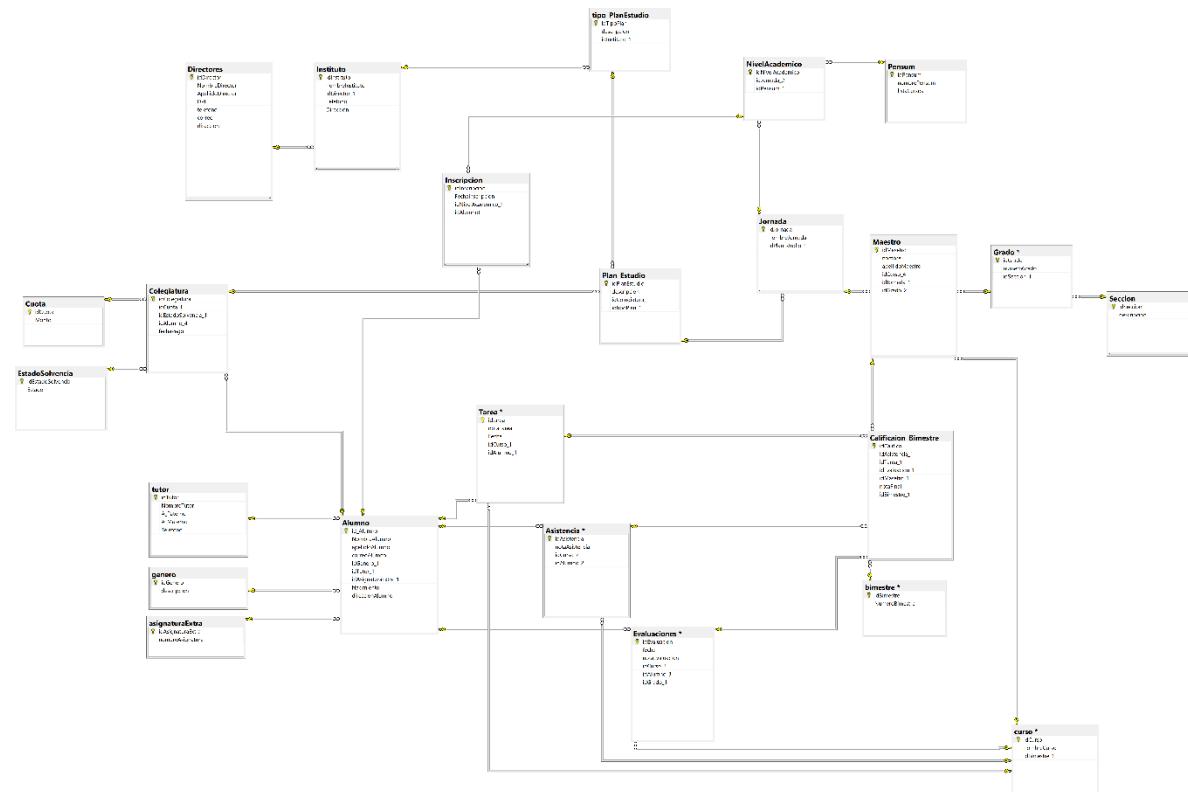
Durante el proceso de normalización de las entidades de los diagramas presentados anteriormente se inicio con la idea de que fuera una base de datos que cumpliera con lo requisitos solicitados siguiendo la normativa, al inicio se colocaron distintas tablas que al final no eran útiles para el sistema y no tenían

una relación concreta no aportaban funcionalidad, así mismo algunas tablas repetidas o que almacenaban datos redundantes.

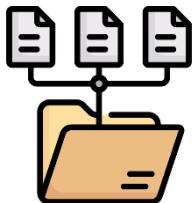


En un inicio se tenían contemplados 26 tablas las cuales se evaluaron y modificaron con el fin de optimizar el funcionamiento de cada tabla.

Como segunda forma Normal se realizo otro diagrama el cual se asemejaba más al diagrama final, pero aun tenía datos que no concordaban con el objetivo del problema.



Al concluir la normalización se crearon 27 tablas las cuales cumplen con la normativa y ayudan a lograr el objetivo fijado. Durante el proceso de normalización se eliminaron tablas como: Dirección, carnet, gradocursos, primaria, básico, diversificado; el cambio se realizó para poder agregar tablas que si influyeran en el buen uso del almacenamiento y puedan colocarse datos



Programas utilizados.

Para poder desarrollar los diagramas utilizamos diferentes herramientas que nos ayudaron a llegar al diagrama final, las cuales son:

SQL Management



SQL Server Management Studio (SSMS para abreviar) es un entorno de desarrollo integrado para administrar cualquier infraestructura SQL. Se utiliza para acceder, administrar, configurar y desarrollar todos los componentes de SQL Server y SQL Database, con este programa se desarrollaron los querys, uno de los diagramas y las consultas

Drawio

Draw.io es un software utilizado para diseñar diagramas de forma gratuita y offline, facilita la integración con múltiples plataformas y programas.

Esta herramienta permite realizar cualquier tipo de diagrama de flujo, diagramas de procesos, organigramas, así como diagramas de red, UML, mapas conceptuales y otros elementos necesarios para realizar un diseño.

Con la ayuda de este software logramos realizar los diagramas entidad relación.

Dia



Es una aplicación que tiene como propósito la creación de diagramas.

Este software está diseñado como un sustituto libre y gratuito. Con esta aplicación se puede desarrollar diferentes tipos de diagramas, ofrece diagramas de entidad-relación, UML, de flujo, de redes, circuitos eléctricos entre muchos más. Para el desarrollo de este proyecto Dia fue utilizado para la creación del diagrama relacional.

CONSULTAS

COMANDOS MAS UTILIZADOS.

1. CREATE Este comando crea un objeto dentro del gestor de base de datos. Puede ser una base de datos, tabla, índice, procedimiento almacenado o vista.

Ejemplo (crear una tabla):

The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, a database named 'notas' is selected. In the center pane, a query window titled 'SQLQuery1.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio' contains the following T-SQL code:

```
CREATE DATABASE notas;
USE notas;
CREATE TABLE alumno (
    idAlumno INT PRIMARY KEY,
    nombreAlumno VARCHAR(50) NOT NULL,
    apellidoAlumno VARCHAR(50) NOT NULL,
    correoAlumno VARCHAR(50) NOT NULL,
    idGenero INT NOT NULL,
    idTutor INT NOT NULL,
    idGrado INT NOT NULL,
    nacimientoAlumno DATE NOT NULL,
    direccionAlumno VARCHAR(50) NOT NULL,
);
CREATE TABLE maestro (
    idMaestro INT PRIMARY KEY IDENTITY,
    nombreMaestro VARCHAR(50) NOT NULL,
    apellidoMaestro VARCHAR(50) NOT NULL,
    correoMaestro VARCHAR(50) NOT NULL,
    telefonoMaestro VARCHAR(50) NOT NULL,
    idCurso INT NOT NULL,
    idJornada INT NOT NULL,
    idGrado INT NOT NULL
);
```

The status bar at the bottom indicates 'Query executed successfully.' and shows the completion time as 2022-10-22T10:37:00.8867451-06:00.

The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, a database named 'notas' is selected. In the center pane, a query window titled 'SQLQuery1.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio' contains the following T-SQL code:

```
CREATE TABLE calificacionBimestral (
    idCalificacionBimestral INT PRIMARY KEY IDENTITY,
    idAsistencia INT NOT NULL,
    idTarea INT NOT NULL,
    idEvaluacion INT NOT NULL,
    idAlumno INT NOT NULL,
    idMaestro INT NOT NULL,
    notaBimestral INT NOT NULL
);

CREATE TABLE calificacionFinalAnual (
    idCalificacionFinalAnual INT PRIMARY KEY IDENTITY,
    idCalificacionBimestral INT NOT NULL,
    idAlumno INT NOT NULL,
    idMaestro INT NOT NULL,
    notaFinalAnual INT NOT NULL
);

ALTER TABLE alumno
ADD FOREIGN KEY (idGenero) REFERENCES genero(idGenero);
ALTER TABLE alumno
ADD FOREIGN KEY (idTutor) REFERENCES alumno(idAlumno);
```

The status bar at the bottom indicates 'Query executed successfully.' and shows the completion time as 2022-10-22T10:37:11.4444917-06:00.

2. SELECT Las consultas de selección se utilizan para indicar al motor de datos que devuelva información de las bases de datos, esta información es devuelta en forma de conjunto de registros. Este conjunto de registros es modificable.

La sintaxis básica de una consulta de selección es:

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

New Query New Item Task List Object Explorer

Execute Stop Refresh Save All Close

notas

SQLQuery30.sql - L... AOH.notas (sa (51))

insert into inscripcion values (99)

-- CONSULTAS

```
select * from alumno
select * from maestro
select * from director
select * from tutor
select * from instituto
select * from inscripcion
select * from colegiatura
select * from cuota
select * from estadoSolvencia
select * from planEstudio
select * from planDiario
select * from planFinsmama
select * from planMadurez
select * from jornada
select * from grado
```

Results Messages

dAlumno	nombreAlumno	apellidoAlumno	correoAlumno	telefonoAlumno	direccionAlumno	dGenero	dIdade	dGrado
1	Patricia	Patricia	pperez@outlook.com	2007-10-15	4891 Balfour Parkway	NULL	NULL	NULL
2	Todd	Upstate	tupscale@outlook.com	2009-08-01	4821 Lighthouse Bay Park	NULL	NULL	NULL
3	Wynne	Bartoszewski	wbartoszewski2@nemsung.com	2003-01-01	1 Walton Park	NULL	NULL	NULL
4	Lanette	Casey	lcasey10@hearthnet.com	2011-10-20	1 Concord Lane	NULL	NULL	NULL
5	Vinnie	Stolley	vstolley9@nhs.uk	2006-12-15	2608 Golf View Terrace	NULL	NULL	NULL
6	Timmie	Kellie	tkellie7@webnode.com	2011-10-09	71735 Dawn Pass	NULL	NULL	NULL
7	Rosemary	Kennet	rkennet8@cam.ac.uk	2012-11-28	650 Coolidge Court	NULL	NULL	NULL
8	Flossi	Hawigarth	fhawigarth7@willian.com	2007-12-09	67 Mandrake Junction	NULL	NULL	NULL

dMaestro	nombreMaestro	apellidoMaestro	correoMaestro	telefonoMaestro	dCurso	dJornada	dOrdo
1	Nesta	Cristoforo	ncristoforo123-reg.co.uk	6248105996	NULL	NULL	NULL
2	Bibi	Greenhouse	bgreenhouse1@theherestore.net	9106739952	NULL	NULL	NULL
3	Mathian	Waind	mwsanderson1@nhs.edu	6108492013	NULL	NULL	NULL
4	Gussey	Neumondale	gnorodinale3@photon.com	2029483052	NULL	NULL	NULL
5	Tracie	Charick	tcharick4@jpmorgan.com	8603698980	NULL	NULL	NULL
6	Torrence	Chertenis	tchertenis5@postenius.com	3227248790	NULL	NULL	NULL
7	Cymbre	Bewington	cbewington6@instagram.com	2177530480	NULL	NULL	NULL

Query executed successfully.

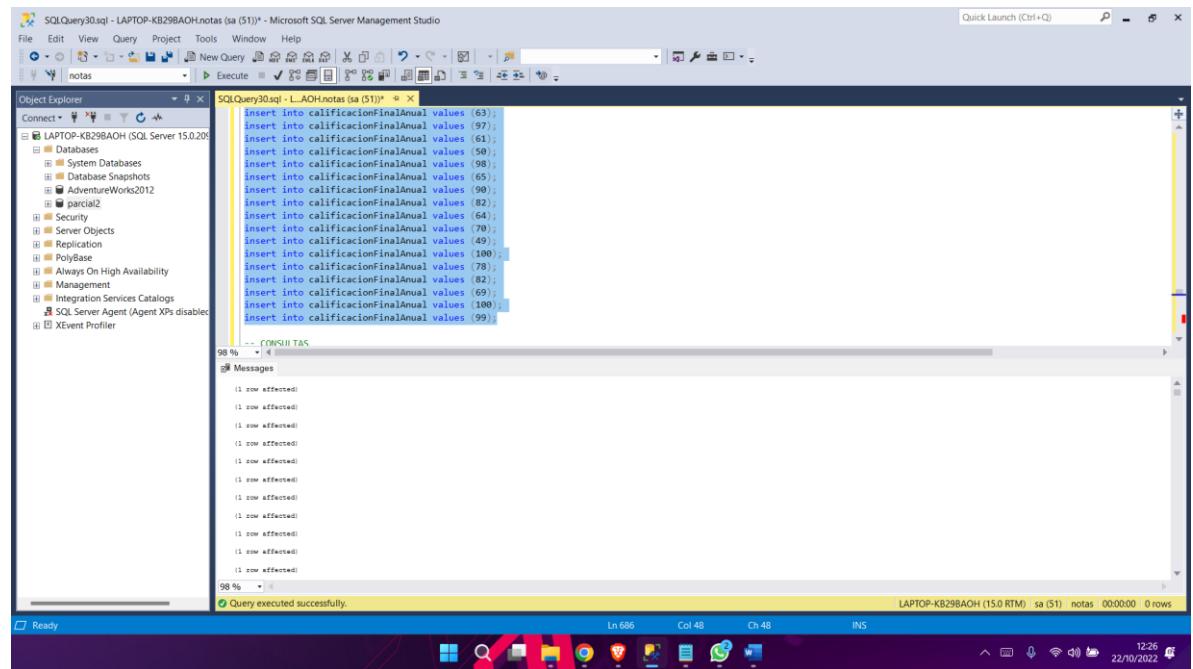
3. 6. ALTER Este comando permite modificar la estructura de un objeto. Se pueden agregar/quitar campos a una tabla, modificar el tipo de un campo, agregar/quitar índices a una tabla, modificar un trigger, etc.

Ejemplo (agregar columna a una tabla):

```
SQLQuery30.sql - L...AUH.notas (sa (51))" ^ A
ALTER TABLE calificacionFinalAnual
ADD FOREIGN KEY (idMaestro) REFERENCES maestro(idMaestro);
```

4. INSERT Una sentencia INSERT de SQL agrega uno o más registros a una (y sólo una) tabla en una base de datos relacional.

Ejemplo:



SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

Execute New Query Object Explorer SQL Server Object Explorer

SQLQuery30sql - L.AOH.notas (sa (51))

```
insert into calificacionFinalAnual values (63);
insert into calificacionFinalAnual values (97);
insert into calificacionFinalAnual values (61);
insert into calificacionFinalAnual values (50);
insert into calificacionFinalAnual values (98);
insert into calificacionFinalAnual values (65);
insert into calificacionFinalAnual values (90);
insert into calificacionFinalAnual values (92);
insert into calificacionFinalAnual values (64);
insert into calificacionFinalAnual values (70);
insert into calificacionFinalAnual values (49);
insert into calificacionFinalAnual values (100);
insert into calificacionFinalAnual values (78);
insert into calificacionFinalAnual values (82);
insert into calificacionFinalAnual values (69);
insert into calificacionFinalAnual values (100);
insert into calificacionFinalAnual values (99);
```

98 %

Messages

```
(1 row affected)
```

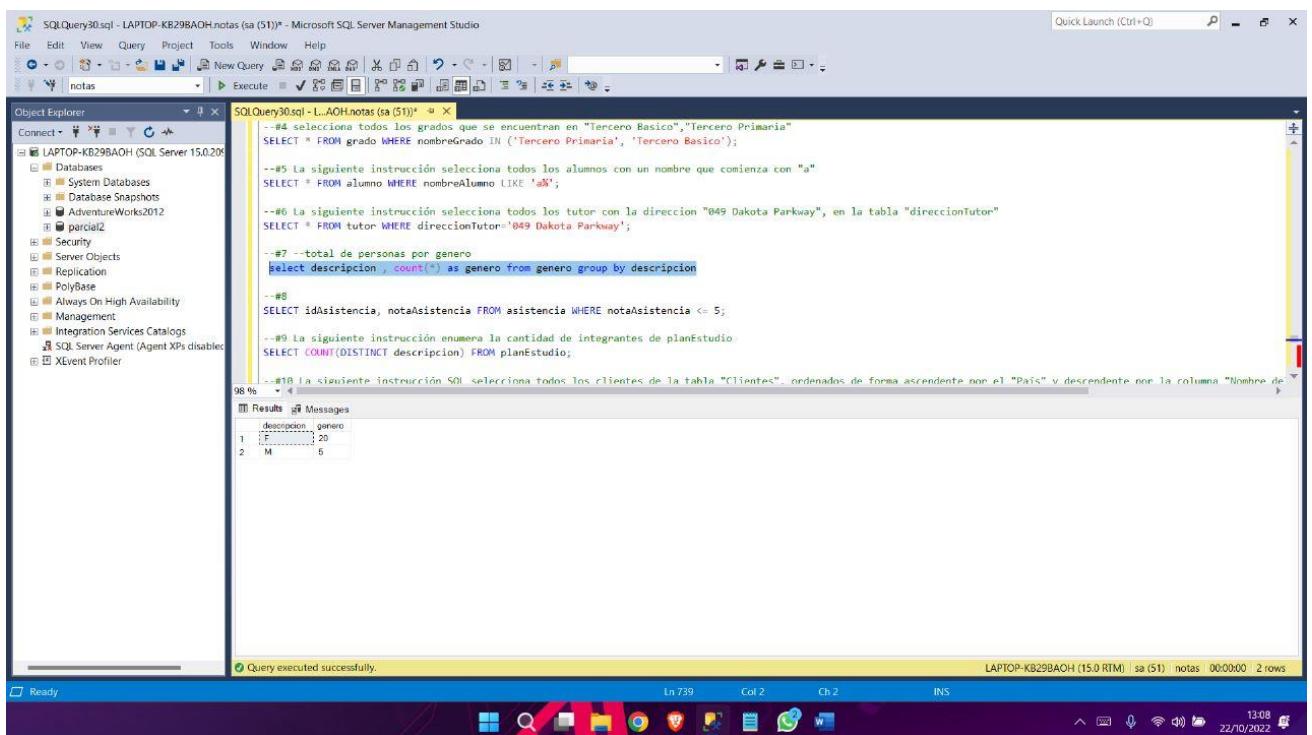
98 %

Query executed successfully.

LAPTOP-KB29BAOH (15.0 RTM) | sa (51) | notas | 00:00:00 | 0 rows

In 686 Col 48 Ch 48 INS

Ready 12:26 22/10/2022



SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

Execute New Query Object Explorer SQL Server Object Explorer

SQLQuery30sql - L.AOH.notas (sa (51))

```
--#4 selecciona todos los grados que se encuentran en "Tercero Basico", "Tercero Primaria"
SELECT * FROM grado WHERE nombreGrado IN ('Tercero Primaria', 'Tercero Basico');

--#5 La siguiente instrucción selecciona todos los alumnos con un nombre que comienza con "a"
SELECT * FROM alumno WHERE nombreAlumno LIKE 'a%';

--#6 La siguiente instrucción selecciona todos los tutor con la dirección "849 Dakota Parkway", en la tabla "direccionTutor"
SELECT * FROM tutor WHERE direccionTutor='849 Dakota Parkway';

--#7 --total de personas por genero
SELECT descripcion , count(*) as genero from genero group by descripcion;

--#8
SELECT idAsistencia , notaAsistencia FROM asistencia WHERE notaAsistencia <= 5;

--#9 La siguiente instrucción enumera la cantidad de integrantes de planEstudio
SELECT COUNT(DISTINCT descripcion) FROM planEstudio;
```

98 %

Results Messages

descripcion	genero
F	20
M	6

Query executed successfully.

LAPTOP-KB29BAOH (15.0 RTM) | sa (51) | notas | 00:00:00 | 2 rows

In 739 Col 2 Ch 2 INS

Ready 13:08 22/10/2022

OPERADORES LOGICOS

13. WHERE

La WHERE cláusula se utiliza para filtrar registros. Se utiliza para extraer solo aquellos registros que cumplen una condición específica.

The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, a database named 'notas' is selected. In the center pane, there is a query window titled 'SQLQuery30.sql - L_AOH.notas (sa (51))'. The query window contains several commented-out SQL statements demonstrating various WHERE clause conditions:

```
--#2 Realiza la consulta en la tabla Evaluacion entre el idAlumno de 1 al 5 con una notaEvaluacion de 40 pts.  
SELECT * FROM evaluacion WHERE (idEvaluacion > 1 AND notaEvaluacion < 5) OR notaEvaluacion = 40;  
  
--#3 Realiza la consulta en la tabla estadoSolvente sobre quienes no están Solvente  
SELECT * FROM estadoSolvencia WHERE NOT Estado = 'Solvente';  
  
--#4 selecciona todos los grados que se encuentran en "Tercero Basico", "Tercero Primaria"  
SELECT * FROM grado WHERE nombreGrado IN ('Tercero Primaria', 'Tercero Basico');  
  
--#5 La siguiente instrucción selecciona todos los alumnos con un nombre que comienza con "a"  
SELECT * FROM alumno WHERE nombreAlumno LIKE 'a%';  
  
--#6 La siguiente instrucción selecciona todos los tutor con la dirección "049 Dakota Parkway", en la tabla "direccionTutor"  
SELECT * FROM tutor  
WHERE direccionTutor='049 Dakota Parkway';  
  
--#7  
SELECT apellidoAlumno, idGenero FROM alumno
```

The results pane shows a table with the following data:

	idAlumno	nombreAlumno	apellidoAlumno	correoAlumno	nacimientoAlumno	direccionAlumno	idGenero	idTutor	idGrado
1	12	Alice	Orteau	aorteaub@cpnpanel.net	2006-08-22	6 Sutteridge Place	NULL	NULL	NULL

At the bottom of the screen, a status bar indicates 'Query executed successfully.' and the system time '12:32 22/10/2022'.

14. DISTINCT

La SELECT DISTINCT declaración se usa para devolver solo valores distintos (diferentes). Dentro de una tabla, una columna suele contener muchos valores duplicados; y, a veces, solo desea enumerar los diferentes valores (distintos).

15. ORDER BY

La ORDER BYpalabra clave se utiliza para clasificar el conjunto de resultados en orden ascendente o descendente. La ORDER BYpalabra clave ordena los registros en orden ascendente de forma predeterminada. Para ordenar los registros en orden descendente, utilice la DESCpalabra clave.

16. DESC

Observamos como introduciendo la cláusula ORDER BY e indicando la columna por la que ordenar, el resultado viene ordenado de forma ascendente (ASC), es decir, de menor a mayor. ¿Y si

queremos ordenar a la inversa, de mayor a menor? Bien, en ese caso se debe indicar que la ordenación es descendente (DESC). Veamos esto tomando como patrón la consulta anterior:

17. ASC

Por tanto si seguido del campo por el que queremos ordenar indicamos ASC, o bien no indicamos nada, la ordenación se hará de forma ascendente, mientras que si indicamos DESC, se hará de forma descendente.

19. AS

El AS comando se usa para cambiar el nombre de una columna o tabla con un alias. Un alias solo existe mientras dura la consulta.

20. VALUES

El VALUES comando especifica los valores de una instrucción INSERT INTO.

21. CASE

El CASE comando que se usa es para crear una salida diferente según las condiciones.

El siguiente SQL pasa por varias condiciones y devuelve un valor cuando se cumple la condición especificada:

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

New Query Execute

notas

--#1 Realiza la consulta en la tabla Evaluacion con las notas > 70 y < 60
SELECT * FROM evaluacion WHERE notaEvaluacion > 70 AND notaEvaluacion < 60;

--#2 Realiza la consulta en la tabla Evaluacion entre el idAlumno de 1 al 5 con una notaEvaluacion de 40 pts.
SELECT * FROM evaluacion WHERE (idAlumno > 1 AND notaEvaluacion < 5) OR notaEvaluacion = 40;

--#3 Realiza la consulta en la tabla estadoSolvente sobre quienes no están Solvente
SELECT * FROM estadoSolvencia WHERE NOT Estado = 'Solvente';

--#4 selecciona todos los grados que se encuentran en "Tercero Basico","Tercero Primaria"
SELECT * FROM grado WHERE nombreGrado IN ('Tercero Primaria', 'Tercero Basico');

--#5 La siguiente instrucción selecciona todos los alumnos con un nombre que comienza con "a"
SELECT * FROM alumno WHERE nombreAlumno LIKE 'a%';

--#6 La siguiente instrucción selecciona todos los tutor con la dirección "049 Dakota Parkway", en la tabla "direccionTutor"

Results Messages

	idEvaluacion	fechaEvaluacion	notaEvaluacion	idCurs	idAlumno	idGrado	idSesion
1	1	2022-10-01	28	NULL	NULL	NULL	NULL
2	2	2022-10-01	13	NULL	NULL	NULL	NULL
3	3	2022-10-01	38	NULL	NULL	NULL	NULL
4	4	2022-10-01	11	NULL	NULL	NULL	NULL
5	5	2022-10-01	41	NULL	NULL	NULL	NULL
6	6	2022-10-01	4	NULL	NULL	NULL	NULL
7	7	2022-10-01	48	NULL	NULL	NULL	NULL
8	8	2022-10-01	6	NULL	NULL	NULL	NULL
9	9	2022-10-01	22	NULL	NULL	NULL	NULL
10	10	2022-10-01	13	NULL	NULL	NULL	NULL
11	11	2022-10-01	41	NULL	NULL	NULL	NULL
12	12	2022-10-01	45	NULL	NULL	NULL	NULL
13	13	2022-10-01	7	NULL	NULL	NULL	NULL
14	14	2022-10-01	48	NULL	NULL	NULL	NULL
15	15	2022-10-01	3	NULL	NULL	NULL	NULL
16	16	2022-10-01	8	NULL	NULL	NULL	NULL
...
19	19	2022-10-01	6	AK.01	AK.01	AK.01	AK.01

Query executed successfully.

LAPTOP-KB29BAOH (15.0 RTM) | sa (51) | notas | 00:00:00 | 25 rows

Ready

Ln 721 Col 1 Ch 1 INS

12:30 22/10/2022

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

New Query Execute

notas

--#1 Realiza la consulta en la tabla Evaluacion con las notas > 70 y < 60
SELECT * FROM evaluacion WHERE notaEvaluacion > 70 AND notaEvaluacion < 60;

--#2 Realiza la consulta en la tabla Evaluacion entre el idAlumno de 1 al 5 con una notaEvaluacion de 40 pts.
SELECT * FROM evaluacion WHERE (idAlumno > 1 AND notaEvaluacion < 5) OR notaEvaluacion = 40;

--#3 Realiza la consulta en la tabla estadoSolvente sobre quienes no están Solvente
SELECT * FROM estadoSolvencia WHERE NOT Estado = 'Solvente';

--#4 selecciona todos los grados que se encuentran en "Tercero Basico","Tercero Primaria"
SELECT * FROM grado WHERE nombreGrado IN ('Tercero Primaria', 'Tercero Basico');

--#5 La siguiente instrucción selecciona todos los alumnos con un nombre que comienza con "a"
SELECT * FROM alumno WHERE nombreAlumno LIKE 'a%';

--#6 La siguiente instrucción selecciona todos los tutor con la dirección "049 Dakota Parkway", en la tabla "direccionTutor"

Results Messages

	idEvaluacion	fechaEvaluacion	notaEvaluacion	idCurs	idAlumno	idGrado	idSesion
1	6	2022-10-01	4	NULL	NULL	NULL	NULL
2	15	2022-10-01	3	NULL	NULL	NULL	NULL
3	21	2022-10-01	40	NULL	NULL	NULL	NULL
4	23	2022-10-01	2	NULL	NULL	NULL	NULL

Query executed successfully.

LAPTOP-KB29BAOH (15.0 RTM) | sa (51) | notas | 00:00:00 | 4 rows

Ready

Ln 724 Col 97 Ch 97 INS

12:30 22/10/2022

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

Execute New Query Find Replace All Open Recent

notas

--#1 Realiza la consulta en la tabla Evaluacion con las notas > 70 y < 60
SELECT * FROM evaluacion WHERE notaEvaluacion > 0 AND notaEvaluacion < 60;

--#2 Realiza la consulta en la tabla Evaluacion entre el idAlumno de 1 al 5 con una notaEvaluacion de 40 pts.
SELECT * FROM evaluacion WHERE (idEvaluacion > 1 AND notaEvaluacion < 5) OR notaEvaluacion = 40;

--#3 Realiza la consulta en la tabla estadoSolvente sobre quienes no están Solvente
SELECT * FROM estadoSolvencia WHERE NOT Estado = 'Solvente';

--#4 selecciona todos los grados que se encuentran en "Tercero Basico", "Tercero Primaria"
SELECT * FROM grado WHERE nombreGrado IN ('Tercero Primaria', 'Tercero Basico');

--#5 La siguiente instrucción selecciona todos los alumnos con un nombre que comienza con "a"
SELECT * FROM alumno WHERE nombreAlumno LIKE 'a%';

--#6 La siguiente instrucción selecciona todos los tutor con la dirección "049 Dakota Parkway", en la tabla "direccionTutor"
SELECT * FROM tutor;

Results Messages

	idEstadoSolvente	Estado
1	2	In solvente
2	5	In solvente
3	8	In solvente
4	11	In solvente
5	14	In solvente
6	17	In solvente
7	20	In solvente
8	23	In solvente

Query executed successfully.

LAPTOP-KB29BAOH (15.0 RTM) sa (51) notas 00:00:00 8 rows

Ready Ln 727 Col 1 Ch 1 INS 12:31 22/10/2022

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

Execute New Query Find Replace All Open Recent

notas

--#1 Realiza la consulta en la tabla Evaluacion con las notas > 70 y < 60
SELECT * FROM evaluacion WHERE notaEvaluacion > 0 AND notaEvaluacion < 60;

--#2 Realiza la consulta en la tabla Evaluacion entre el idAlumno de 1 al 5 con una notaEvaluacion de 40 pts.
SELECT * FROM evaluacion WHERE (idEvaluacion > 1 AND notaEvaluacion < 5) OR notaEvaluacion = 40;

--#3 Realiza la consulta en la tabla estadoSolvente sobre quienes no están Solvente
SELECT * FROM estadoSolvencia WHERE NOT Estado = 'Solvente';

--#4 selecciona todos los grados que se encuentran en "Tercero Basico", "Tercero Primaria"
SELECT * FROM grado WHERE nombreGrado IN ('Tercero Primaria', 'Tercero Basico');

--#5 La siguiente instrucción selecciona todos los alumnos con un nombre que comienza con "a"
SELECT * FROM alumno WHERE nombreAlumno LIKE 'a%';

--#6 La siguiente instrucción selecciona todos los tutor con la dirección "049 Dakota Parkway", en la tabla "direccionTutor"
SELECT * FROM tutor;

Results Messages

	idGrado	nombreGrado	idSeccion
1	3	Tercero Primario	NULL
2	9	Tercero Basico	NULL

Query executed successfully.

LAPTOP-KB29BAOH (15.0 RTM) sa (51) notas 00:00:00 2 rows

Ready Ln 730 Col 1 Ch 1 INS 12:32 22/10/2022

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

New Query Execute

--#2 Realiza la consulta en la tabla Evaluacion entre el idAlumno de 1 al 5 con una notaEvaluacion de 40 pts.
SELECT * FROM evaluacion WHERE (idEvaluacion > 1 AND notaEvaluacion < 5) OR notaEvaluacion = 40;

--#3 Realiza la consulta en la tabla estadoSolvente sobre quienes no están Solvente
SELECT * FROM estadoSolvente WHERE NOT Estado = 'Solvente';

--#4 selecciona todos los grados que se encuentran en "Tercero Basico","Tercero Primaria"
SELECT * FROM grado WHERE nombreGrado IN ('Tercero Primaria', 'Tercero Basico');

--#5 La siguiente instrucción selecciona todos los alumnos con un nombre que comienza con "a"
SELECT * FROM alumno WHERE nombreAlumno LIKE 'a%';

--#6 La siguiente instrucción selecciona todos los tutor con la dirección "049 Dakota Parkway", en la tabla "direccionTutor"
SELECT * FROM tutor
WHERE direccionTutor='049 Dakota Parkway';

--#7
SELECT apellidoAlumno, idGenero FROM alumno

Results Messages

idAlumno	nombreAlumno	apellidoAlumno	correoAlumno	nacimientoAlumno	direccionAlumno	idGenero	idTutor	idGrado
1	Alice	Oteau	aoteaub@spulan.net	2006-06-22	6 Suttleidge Place	NULL	NULL	NULL

Query executed successfully.

LAPTOP-KB29BAOH (15.0 RTM) sa (51) notas 00:00:00 1 rows

Ready Ln 733 Col 1 Ch 1 INS 12:32 22/10/2022

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

New Query Execute

--#5 La siguiente instrucción selecciona todos los alumnos con un nombre que comienza con "a"
SELECT * FROM alumno WHERE nombreAlumno LIKE 'a%';

--#6 La siguiente instrucción selecciona todos los tutor con la dirección "049 Dakota Parkway", en la tabla "direccionTutor"
SELECT * FROM tutor WHERE direccionTutor='049 Dakota Parkway';

--#7
SELECT apellidoAlumno, idGenero FROM alumno
WHERE idGenero > 2;

--#8
SELECT idAsistencia, notaAsistencia FROM asistencia
WHERE notaAsistencia <= 5;

--#9 La siguiente instrucción enumera la cantidad de integrantes de planEstudio
SELECT COUNT(DISTINCT descripcion) FROM planEstudio;

--#10 La siguiente instrucción SQL selecciona todos los clientes de la tabla "Clientes", ordenados de forma ascendente por el "Pais" y descendente por la columna "Nombre de...

--#11 La siguiente instrucción SQL selecciona todos los clientes de la tabla "Clientes", ordenados de forma ascendente por el "Pais" y descendente por la columna "Nombre de..."
SELECT * FROM alumno

Results Messages

idTutor	nombreTutor	apellidoTutor	direccionTutor	telefonoTutor	tipoParentesco	idAlumno
1	Leila	Woolfson	049 Dakota Parkway	7419536194	Papa	NULL

Query executed successfully.

LAPTOP-KB29BAOH (15.0 RTM) sa (51) notas 00:00:00 1 rows

Ready Ln 736 Col 1 Ch 1 INS 12:32 22/10/2022

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

New Query Execute

notas

--#6 La siguiente instrucción selecciona todos los tutor con la dirección "049 Dakota Parkway", en la tabla "direccionTutor"
SELECT * FROM tutor WHERE direccionTutor='049 Dakota Parkway';
--#7
--#8
--#9 La siguiente instrucción enumera la cantidad de integrantes de planEstudio
SELECT COUNT(DISTINCT descripción) FROM planEstudio;
--#10 La siguiente instrucción SQL selecciona todos los clientes de la tabla "Clientes", ordenados de forma ascendente por el "País" y descendente por la columna "Nombre de
SELECT * FROM alumno
ORDER BY nacimientoAlumno ASC, direcciónAlumno DESC;
--#11 descendente
select nombreAlumno,apellidoAlumno,correoAlumno
from alumno

Results Messages

	idAsistencia	notaAsistencia
1	3	2
2	7	5
3	9	5
4	10	4
5	16	2
6	19	3
7	20	5
8	23	4

Query executed successfully.

LAPTOP-KB29BAOH (15.0 RTM) sa (51) notas 00:00:00 8 rows

Ready Ln 741 Col 79 Ch 79 INS 12:37 22/10/2022

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

New Query Execute

notas

--#6 La siguiente instrucción selecciona todos los tutor con la dirección "049 Dakota Parkway", en la tabla "direccionTutor"
SELECT * FROM tutor WHERE direccionTutor='049 Dakota Parkway';
--#7
--#8
--#9 La siguiente instrucción enumera la cantidad de integrantes de planEstudio
SELECT COUNT(DISTINCT descripción) FROM planEstudio;
--#10 La siguiente instrucción SQL selecciona todos los clientes de la tabla "Clientes", ordenados de forma ascendente por el "País" y descendente por la columna "Nombre de
SELECT * FROM alumno
ORDER BY nacimientoAlumno ASC, direcciónAlumno DESC;
--#11 descendente
select nombreAlumno,apellidoAlumno,correoAlumno
from alumno

Results Messages

	idAsistencia	notaAsistencia
1	3	2
2	7	5
3	9	5
4	10	4
5	16	2
6	19	3
7	20	5
8	23	4

Query executed successfully.

LAPTOP-KB29BAOH (15.0 RTM) sa (51) notas 00:00:00 8 rows

Ready Ln 741 Col 79 Ch 79 INS 12:37 22/10/2022

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

```
--#8
SELECT idAsistencia, notaAsistencia FROM asistencia WHERE notaAsistencia <= 5;

--#9 La siguiente instrucción enumera la cantidad de integrantes de planEstudio
SELECT COUNT(DISTINCT descripción) FROM planEstudio;

--#10 La siguiente instrucción SQL selecciona todos los clientes de la tabla "Clientes", ordenados de forma ascendente por el "País" y descendente por la columna "Nombre de
SELECT * FROM alumno ORDER BY nacimientoAlumno ASC, direcciónAlumno DESC;

--#11 descendente
select nombreAlumno, apellidoAlumno, correoAlumno
from alumno
order by correoAlumno desc

--#12 ascendente
select nombreAlumno, apellidoAlumno, correoAlumno
from alumno
order by correoAlumno asc
```

alumno	nombreAlumno	apellidoAlumno	correoAlumno	nacimientoAlumno	direcciónAlumno	idAsistencia	idUser	idRadio	
1	Wynne	Barbosawski	wbarbosawski2@hansung.com	2003-01-01	1 Walton Park	NULL	NULL	NULL	
2	Tanya	Tongs	tongso@nature.com	2004-05-04	6 Foster Hill	NULL	NULL	NULL	
3	Danni	Kyte	dytter@npr.org	2004-10-06	63682 8th Alley	NULL	NULL	NULL	
4	Patti	Keast	pkeast@wix.com	2005-02-26	89 Express Terrace	NULL	NULL	NULL	
5	Kleon	Mucklo	kmucklom@weather.com	2005-02-27	8 Southridge Pass	NULL	NULL	NULL	
6	Alice	Oreata	aoreata@spacetel.net	2005-08-26	6 Sutherland Plaza	NULL	NULL	NULL	
7	Gwendelle	Holroyden	gholroyden@lynnews.com	2006-01-01	34 Hazelwood Junction	NULL	NULL	NULL	
8	Vincent	Bailey	vbailey4@tiny.cc	2006-12-15	2600 Gulf View Terrace	NULL	NULL	NULL	
9	Eula	Sotoe	esoper@voda.gov	2007-04-08	2 Stephen Road	NULL	NULL	NULL	
10	Shirlene	Cavendish	scavendish@yandex.ru	2007-08-02	23002 Sunfield Crossing	NULL	NULL	NULL	
11	1	Gustavus	Mayet	gmayne@facebook.com	2007-10-11	569 Sutherland Parkway	NULL	NULL	NULL
12	8	Flossi	hwylgarth@telillian.com	2007-12-09	67 Mandrake Junction	NULL	NULL	NULL	
13	21	Hinda	honiindsk@mh.com.au	2008-01-30	2 Golf Course Road	NULL	NULL	NULL	
14	Rudolf	Burren	rburnea@halite.com	2008-04-13	20559 Thivier Place	NULL	NULL	NULL	
15	24	Ogert	ogarte@archive.org	2008-12-21	5 Olive Hill	NULL	NULL	NULL	
16	Tilda	Upsale	upsale1@businesswise.com	2009-08-01	4241 Lighthouse Bay Park	NULL	NULL	NULL	
17	n	Blundell	blundell@tiny.cc	2009-08-08	8420 4th Street	NULL	NULL	NULL	

Query executed successfully.

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

```
--#8
SELECT idAsistencia, notaAsistencia FROM asistencia WHERE notaAsistencia <= 5;

--#9 La siguiente instrucción enumera la cantidad de integrantes de planEstudio
SELECT COUNT(DISTINCT descripción) FROM planEstudio;

--#10 La siguiente instrucción SQL selecciona todos los clientes de la tabla "Clientes", ordenados de forma ascendente por el "País" y descendente por la columna "Nombre de
SELECT * FROM alumno ORDER BY nacimientoAlumno ASC, direcciónAlumno DESC;

--#11 descendente
select nombreAlumno, apellidoAlumno, correoAlumno
from alumno
order by correoAlumno desc

--#12 ascendente
select nombreAlumno, apellidoAlumno, correoAlumno
from alumno
order by correoAlumno asc

--#13 Estudiantes que tienen asignaturaExtra de IDIOMA MAYA
select * from asignaturaExtra
```

nombreAlumno	apellidoAlumno	correoAlumno
1 Wynne	Barbosawski	wbarbosawski2@hansung.com
2 Robin	Bailey	vbailey4@tiny.cc
3 Tidy	Upsale	upsale1@businesswise.com
4 Tobin	Trinbey	trinbey@hastels.com
5 Tanya	Tongs	tongso@nature.com
6 Timmo	Kelbell	tkelbel5@webnode.com
7 Shirlene	Cavendish	scavendish@yandex.ru
8 Rosemary	Kennet	rkenett@cam.ac.uk
9 Rudolf	Burren	rburnea@halite.com
10 Pauline	Blundell	blundell@tiny.cc
11 Patti	Keast	pkeast@wix.com
12 Orla	Gart	ogarte@archive.org
13 Lindy	Ollie	lollet@dropbox.com
14 Lanette	Casey	lcasey2@behance.net
15 Kleon	Mucklo	kmucklom@weather.com
16 Kerby	Coggings	kcoggings@yume.com
17	Elspeth	elspeth@tiny.cc

Query executed successfully.

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

```
--#9 La siguiente instrucción enumera el cantidad de integrantes de planEstudio
SELECT COUNT(DISTINCT descripcion) FROM planEstudio;

--#10 La siguiente instrucción SQL selecciona todos los clientes de la tabla "Clientes", ordenados de forma ascendente por el "País" y descendente por la columna "Nombre de
SELECT * FROM alumno ORDER BY nacimientoAlumno ASC, direccionAlumno DESC;

--#11 descendente
select nombreAlumno,apellidoAlumno,correoAlumno from alumno order by correoAlumno desc
column correoAlumno varchar,not null

--#12 ascendente
select nombreAlumno,apellidoAlumno,correoAlumno from alumno order by correoAlumno asc

--#13 Estudiantes que tienen asignaturaExtra de IDIOMA MAYA
select * from asignaturaExtra
where idAsignaturaExtra in(select idAsignaturaExtra from asignaturaExtra where nombreAsignatura='IDIOMA MAYA')

--#14 Realiza la consulta de la cuota mas elevada
select * from cuota
where montoCuota = (select MAX (montoCuota) from cuota)

--#15 Cuentas calificaciones existen entre 80 y 100 pts en la tabla calificacionFinalAnual
select * from calificacionFinalAnual
where notaFinalAnual between 80 and 100;

--SUBCONSULTAS
--#1 Lo que se esta haciendo es desde los alumnos se obtienen en los nombres de los alumnos del menor al promedio de la jornada
```

Results Messages

nombreAlumno	apellidoAlumno	correoAlumno
1	Alejandro	arhevald@open.net
2	Davni	dlytree@msn.org
3	Doreen	dtremoulet@scrbd.com
4	Eula	esparre@fuda.gov
5	Flossi	fshawpath@telian.com
6	Farand	frigodido@rediff.com
7	Gabrielle	ghenryson@ndbynews.com
8	Hannicus	gjernand@comcast.net
9	Hinda	hennrich@vbn.com.au
10	Kerly	kmcogging@vines.com
11	Kleon	kmucklow@weather.com
12	Laney	lcasey@behance.net
13	Lindy	lollett@dropbox.com
14	Offella	ogant@archive.org
15	Patti	pkaatj@wjs.com
16	Reidelle	rstanley@newswire.com
17	Ron	rwilcox@lawn.com

Query executed successfully.

LAPTOP-KB29BAOH (15.0 RTM) sa (51) notas 00:00:00 25 rows

Ready Ln 753 Col 1 Ch 1 INS 12:38 22/10/2022

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

```
--#11 descendente
select nombreAlumno,apellidoAlumno,correoAlumno from alumno order by correoAlumno desc

--#12 ascendente
select nombreAlumno,apellidoAlumno,correoAlumno from alumno order by correoAlumno asc

--#13 Estudiantes que tienen asignaturaExtra de IDIOMA MAYA
select * from asignaturaExtra
where idAsignaturaExtra in(select idAsignaturaExtra from asignaturaExtra where nombreAsignatura='IDIOMA MAYA');

--#14 Realiza la consulta de la cuota mas elevada
select * from cuota
where montoCuota = (select MAX (montoCuota) from cuota)

--#15 Cuentas calificaciones existen entre 80 y 100 pts en la tabla calificacionFinalAnual
select * from calificacionFinalAnual
where notaFinalAnual between 80 and 100;

--SUBCONSULTAS
--#1 Lo que se esta haciendo es desde los alumnos se obtienen en los nombres de los alumnos del menor al promedio de la jornada
```

Results Messages

idAsignaturaExtra	nombreAsignatura
1	IDIOMA MAYA
2	IDIOMA MAYA
3	IDIOMA MAYA
4	IDIOMA MAYA
5	IDIOMA MAYA
6	IDIOMA MAYA
7	IDIOMA MAYA
8	IDIOMA MAYA
9	IDIOMA MAYA
10	IDIOMA MAYA
11	IDIOMA MAYA
12	IDIOMA MAYA
13	IDIOMA MAYA
14	IDIOMA MAYA
15	IDIOMA MAYA
16	IDIOMA MAYA
17	IDIOMA MAYA
18	IDIOMA MAYA
19	IDIOMA MAYA
20	IDIOMA MAYA
21	IDIOMA MAYA
22	IDIOMA MAYA
23	IDIOMA MAYA
24	IDIOMA MAYA
25	IDIOMA MAYA

Query executed successfully.

LAPTOP-KB29BAOH (15.0 RTM) sa (51) notas 00:00:00 15 rows

Ready Ln 757 Col 1 Ch 1 INS 12:38 22/10/2022

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

```
--#11 descendente
select nombreAlumno, apellidoAlumno,correoAlumno from alumno order by correoAlumno desc

--#12 ascendente
select nombreAlumno, apellidoAlumno,correoAlumno from alumno order by correoAlumno asc

--#13 Estudiantes que tienen asignaturaExtra de IDIOMA MAYA
select * from asignaturaExtra where idAsignaturaExtra in(select idAsignaturaExtra from asignaturaExtra where nombreAsignatura='IDIOMA MAYA')

--#14 Realiza la consulta de la cuota mas elevada
select * from cuota where montoCuota = (select MAX(montoCuota) from cuota)

--#15 Cuantas calificaciones existen entre 80 y 100 pts en la tabla calificacionFinalAnual
select * from calificacionFinalAnual
where notaFinalAnual between 80 and 100;

--SUBCONSULTAS
--#1 Lo que se esta haciendo es desde los alumnos se obtienen en los nombres de los alumnos del menor al promedio de la jornada
select nombreAlumno, idGrado from alumno where gradoAlumno < (select avg(jornada) from jornadaAlumno)
```

Results

idCuota	montoCuota
3	300

Query executed successfully.

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

```
--#12 ascendente
select nombreAlumno, apellidoAlumno,correoAlumno from alumno order by correoAlumno asc

--#13 Estudiantes que tienen asignaturaExtra de IDIOMA MAYA
select * from asignaturaExtra where idAsignaturaExtra in(select idAsignaturaExtra from asignaturaExtra where nombreAsignatura='IDIOMA MAYA')

--#14 Realiza la consulta de la cuota mas elevada
select * from cuota where montoCuota = (select MAX(montoCuota) from cuota)

--#15 Cuantas calificaciones existen entre 80 y 100 pts en la tabla calificacionFinalAnual
select * from calificacionFinalAnual where notaFinalAnual between 80 and 100;

--SUBCONSULTAS
--#1 Lo que se esta haciendo es desde los alumnos se obtienen en los nombres de los alumnos del menor al promedio de la jornada
select nombreAlumno, idGrado from alumno where gradoAlumno < (select avg(jornada) from jornadaAlumno)

--#2 Lo que se esta haciendo es desde los alumnos se obtienen en los nombres de los alumnos mayor al promedio de la jornada
select nombreAlumno, seleccion from alumno where gradoAlumno > (select avg(jornada) from jornadaAlumno)
```

Results

idCalificacionFinalAnual	notaFinalAnual	idCalificadorBimestral	idAlumno	idMastro
1	97	NULL	NULL	NULL
2	98	NULL	NULL	NULL
3	90	NULL	NULL	NULL
4	82	NULL	NULL	NULL
5	100	NULL	NULL	NULL
6	82	NULL	NULL	NULL
7	100	NULL	NULL	NULL
8	99	NULL	NULL	NULL

Query executed successfully.

SUBCONSULTAS

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

```
--#1 Lo que se esta haciendo es desde los alumnos se obtienen en los nombres de los alumnos del menor al promedio de la jornada
select nombreAlumno, idGrado from alumno where gradoAlumno < (select avg(jornada) from jornadaAlumno);

--#2 Lo que se esta haciendo es desde los alumnos se obtienen en los nombres de los alumnos mayor al promedio de la jornada
SELECT * from alumno where idAlumno > (select AVG(idGrado) from grado);

--#3 Se selecciona todos los alumnos que esten asignados en el curso de matematica
SELECT nombreAlumno from alumno where nombreAlumno > all (select idCurso from curso where nombreCurso = 'matematica');

--#4 Se selecciona algunos alumnos que esten asignados en el curso de matematica
select * from asignaturas where alumnos > any (select asignaturas from Alumnos where seccion='matematica');

--#5 Se mostrara los alumnos que tengan una asistencia de 20 veces
select nombreAlumno from alumno where asistencia in (select grado from asistencia where asistencia >20);

--JOINS
```

Results [Messages]

idAlumno	nombreAlumno	apellidoAlumno	correoAlumno	nacimientoAlumno	direccionAlumno	idGenero	idTutor	idGrado
1	Rosemary	Kennet	rkenett@cam.ac.uk	2012-11-28	650 Coundige Court	NULL	NULL	NULL
2	Flossi	Hawigarth	fhawigarth@newellan.com	2007-12-09	67 Mandrake Junction	NULL	NULL	NULL
3	Rozie	Blankley	rblankley@newswine.com	2009-08-20	5 5th Hill	NULL	NULL	NULL
4	Lindy	Oller	oller@scopbox.com	2011-04-13	94177 Stephen Park	NULL	NULL	NULL
5	Rudolf	Buren	rburner@ilate.com	2009-08-20	20559 Thorne Place	NULL	NULL	NULL
6	Alice	Asotthakone	aasotthakone@syndex.net	2006-09-22	8 Edgebridge Place	NULL	NULL	NULL
7	Eula	Spree	espre@eude.org	2007-04-08	2 Stephen Road	NULL	NULL	NULL
8	Shirlene	Cavendish	scavendish@syndex.eu	2007-05-02	2302 Sunfield Crossing	NULL	NULL	NULL
9	Danni	Kyter	dkyter@nrg.org	2004-10-10	6392 28 Alley	NULL	NULL	NULL
10	Kerdy	Cogings	koggings@vines.com	2009-10-21	27345 Bowman Way	NULL	NULL	NULL
11	Tanja	Tongs	tongs@nature.com	2004-05-04	6 Forster Hill	NULL	NULL	NULL
12	Gabrielle	Hempsell	ghempsell@syndexnews.com	2006-09-03	34 Namakagon Junction	NULL	NULL	NULL
13	Tobin	Trimby	ttrimby@scopbox.com	2010-01-10	0 Boktong Pass	NULL	NULL	NULL
14	Pete	Platt	pplatt@vines.com	2005-02-26	48 Loxton Trace	NULL	NULL	NULL
15	Henda	Ionnida	hionnida@nh.com.au	2008-01-30	2 Gold Course Road	NULL	NULL	NULL
16	Dorsene	Tremoulet	dtremoulet@scibit.com	2012-01-12	231 Myrtle Circle	NULL	NULL	NULL
17	Tamara	Muskete	tmuskete@cam.ac.uk	2006-01-27	9 Glendale Green	NULL	NULL	NULL

Query executed successfully.

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

```
--#1 Se realiza la consulta de la cuota mas elevada
select * from cuota where montoCuota = (select MAX (montoCuota) from cuota)

--#15 Cuantas calificaciones existen entre 80 y 100 pts en la tabla calificacionFinalAnual
select * from calificacionFinalAnual where notaFinalAnual between 80 and 100;

--SUBCONSULTAS

--#1 Lo que se esta haciendo es desde los alumnos se obtienen en los nombres de los alumnos del menor al promedio de la jornada
select nombreAlumno, idGrado from alumno where gradoAlumno < (select avg(jornada) from jornadaAlumno);

--#2 Lo que se esta haciendo es desde los alumnos se obtienen en los nombres de los alumnos mayor al promedio de la jornada
SELECT * from alumno where gradoAlumno > (select avg(jornada) from jornadaAlumno);

--#3 Se selecciona todos los alumnos que esten asignados en el curso de matematica
SELECT nombreAlumno from alumno where nombreAlumno > all (select idCurso from curso where nombreCurso = 'matematica');

--#4 Se selecciona algunos alumnos que esten asignados en el curso de matematica
select * from asignaturas where alumnos > any (select asignaturas from Alumnos where seccion='matematica');
```

Results [Messages]

nombreAlumno
Gustavus
Tidy
Erica
Landrie
Vinnie
Timmy
Rosemary
Flossi
Rozie
Lindy
Asotthakone
Eula
Shirlene
Danni
Kerdy
Tanja

Query executed successfully.

SQLQuery30.sql - LAPTOP-KB29BAOH.notas (sa (51)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

Execute New Query

notas

Object Explorer

LAPTOP-KB29BAOH (SQL Server 15.0.2000.5)

Databases

System Databases

Database Snapshots

AdventureWorks2012

parcial

Security

Server Objects

Replication

PolyBase

Always On High Availability

Management

Integration Services Catalogs

SQL Server Agent (Agent XPs disabled)

XEvent Profiler

--#5
select *
from seccion secn
right join alumno alm
on secn.idseccion = alm.idAlumno
where alm.idAlumno = null

SELECT alumno.idAlumno, alumno.nombreAlumno, maestro.nombreMaestro FROM alumno INNER JOIN maestro ON alumno.idAlumno = maestro.idMaestro

Results Messages

idAlumno	nombreAlumno	nombreMaestro
1	Gustavo	Natalia
2	Tilly	Bibi
3	Wynne	Mathan
4	Lanette	Gussy
5	Vinnie	Travis
6	Timme	Torrance
7	Rosemary	Cymbre
8	Fernando	Joellen
9	Rossie	Josue
10	Lindy	Flor
11	Rudolf	Pandora
12	Alice	Vaughan
13	Eula	Didi
14	Shirlene	Linnea
15	Doreen	Gardner
16	Kelby	Binne
17	Tammie	Wilma

Query executed successfully.

LAPTOP-KB29BAOH (15.0 RTM) | sa (51) | notas | 00:00:00 | 25 rows

Ready Ln 820 Col 1 Ch 1 INS

13:05 22/10/2022

```
CREATE DATABASE notas;
```

```
USE notas
```

```
CREATE TABLE alumno (
    idAlumno INT PRIMARY KEY IDENTITY,
    nombreAlumno VARCHAR(50) not null,
    apellidoAlumno VARCHAR(50) not null,
    correoAlumno VARCHAR(50) not null,
    nacimientoAlumno DATE not null,
    direccionAlumno VARCHAR(50) not null,
);
```

```
CREATE TABLE maestro (
    idMaestro INT PRIMARY KEY IDENTITY,
    nombreMaestro VARCHAR(50) not null,
    apellidoMaestro VARCHAR(50) not null,
    correoMaestro VARCHAR(50) not null,
    telefonoMaestro VARCHAR(50) not null,
```

);

```
create table director (
    idDirector INT PRIMARY KEY IDENTITY,
    nombreDirector VARCHAR(50) not null,
    apellidoDirector VARCHAR(50) not null,
    dpi VARCHAR(20) not null,
    telefonoDirector VARCHAR(50) not null,
    correoDirector VARCHAR(50) not null,
    direccionDirector VARCHAR(50) not null
);
```

```
create table tutor (
    idTutor INT PRIMARY KEY IDENTITY,
    nombreTutor VARCHAR(50) not null,
    apellidoTutor VARCHAR(50) not null,
    direccionTutor VARCHAR(50) not null,
    telefonoTutor VARCHAR(50) not null,
    tipoParentesco VARCHAR(6) not null,
```

);

```
create table instituto (
    idInstituto INT PRIMARY KEY IDENTITY,
    nombreInstituto VARCHAR(50) not null,
    telefonoInstituto NUMERIC(15) not null,
    direccionInstituto VARCHAR(650) not null,
);
```

```
create table inscripcion (
    idInscripcion INT PRIMARY KEY IDENTITY,
    fechaInscripcion DATE not null,
);
```

```
CREATE TABLE colegiatura (
    idColegiatura INT PRIMARY KEY IDENTITY,
    fechaPago DATE not null
);
```

```
CREATE TABLE cuota (
    idCuota INT PRIMARY KEY IDENTITY,
    montoCuota VARCHAR(50) not null
);
```

```
CREATE TABLE estadoSolvencia (
    idEstadoSolvencia INT PRIMARY KEY IDENTITY,
    estado VARCHAR(50) not null
);
```

```
CREATE TABLE planEstudio (
    idPlanEstudio INT PRIMARY KEY IDENTITY,
    descripcion VARCHAR(60) not null,
);
```

```
CREATE TABLE planDiario (
    idPlanDiario INT PRIMARY KEY IDENTITY,
);
```

```
CREATE TABLE planFinSemana (
    idPlanFinSemana INT PRIMARY KEY IDENTITY,
);
```

```
CREATE TABLE planMadurez (
    idPlanMadurez INT PRIMARY KEY IDENTITY,
);
```

```
CREATE TABLE jornada (
    idJornada INT PRIMARY KEY IDENTITY,
    nombreJornada VARCHAR(50) not null,
);
```

```
CREATE TABLE grado (
    idGrado INT PRIMARY KEY IDENTITY,
    nombreGrado VARCHAR(50) not null,
);
```

```
CREATE TABLE seccion (
```

```
    idSeccion INT PRIMARY KEY IDENTITY,  
    nombreSeccion VARCHAR(1) not null  
);
```

```
CREATE TABLE curso (  
    idCurso INT PRIMARY KEY IDENTITY,  
    nombreCurso VARCHAR(200) not null,  
);
```

```
CREATE TABLE bimestre (   
    idBimestre INT PRIMARY KEY IDENTITY,  
    nombreBimestre VARCHAR(50) not null,  
);
```

```
CREATE TABLE pensum (   
    idPensum INT PRIMARY KEY IDENTITY,  
    listaCursos VARCHAR(120) not null  
);
```

```
CREATE TABLE nivelAcademico (
    idNivelAcademico INT PRIMARY KEY IDENTITY,
);
```

```
CREATE TABLE genero (
    idGenero INT PRIMARY KEY IDENTITY,
    descripcion VARCHAR(10) not null,
);
```

```
CREATE TABLE asignaturaExtra (
    idAsignaturaExtra INT PRIMARY KEY IDENTITY,
    nombreAsignatura VARCHAR(50) not null,
);
```

```
CREATE TABLE tarea (
    idTarea INT PRIMARY KEY IDENTITY,
    notaTarea INT not null,
    fechaTarea DATE not null,
);
```

```
CREATE TABLE asistencia (
    idAsistencia INT PRIMARY KEY IDENTITY,
    notaAsistencia INT not null,
);
```

```
CREATE TABLE evaluacion (
    idEvaluacion INT PRIMARY KEY IDENTITY,
    fechaEvaluacion DATE not null,
    notaEvaluacion INT not null,
);
```

```
CREATE TABLE calificacionBimestral (
    idCalificacionBimestral INT PRIMARY KEY IDENTITY,
    notaBimestral INT not null,
);
```

```
CREATE TABLE calificacionFinalAnual (
    idCalificacionFinalAnual INT PRIMARY KEY IDENTITY,
```

```
    notaFinalAnual INT not null  
);  
  
ALTER TABLE alumno ADD idGenero INT;  
ALTER TABLE alumno ADD idTutor INT;  
ALTER TABLE alumno ADD idGrado INT;  
ALTER TABLE alumno  
ADD FOREIGN KEY (idGenero) REFERENCES genero(idGenero);  
ALTER TABLE alumno  
ADD FOREIGN KEY (idTutor) REFERENCES tutor(idTutor);  
ALTER TABLE alumno  
ADD FOREIGN KEY (idGrado) REFERENCES grado(idGrado);  
  
ALTER TABLE maestro ADD idCurso INT;  
ALTER TABLE maestro ADD idJornada INT;  
ALTER TABLE maestro ADD idGrado INT;  
ALTER TABLE maestro  
ADD FOREIGN KEY (idCurso) REFERENCES curso(idCurso);  
ALTER TABLE maestro
```

ADD FOREIGN KEY (idJornada) REFERENCES
jornada(idJornada);

ALTER TABLE maestro

ADD FOREIGN KEY (idGrado) REFERENCES grado(idGrado);

ALTER TABLE tutor ADD idAlumno INT;

ALTER TABLE tutor

ADD FOREIGN KEY (idAlumno) REFERENCES
alumno(idAlumno);

ALTER TABLE instituto ADD idDirector INT;

ALTER TABLE instituto

ADD FOREIGN KEY (idDirector) REFERENCES
director(idDirector);

ALTER TABLE inscripcion ADD idAlumno INT;

ALTER TABLE inscripcion ADD idNivelAcademico INT;

ALTER TABLE inscripcion

ADD FOREIGN KEY (idAlumno) REFERENCES
alumno(idAlumno);

ALTER TABLE inscripcion

ADD FOREIGN KEY (idNivelAcademico) REFERENCES
nivelAcademico(idNivelAcademico);

ALTER TABLE colegiatura ADD idCuota INT;

ALTER TABLE colegiatura ADD idEstadoSolvencia INT;

ALTER TABLE colegiatura ADD idAlumno INT;

ALTER TABLE colegiatura

ADD FOREIGN KEY (idCuota) REFERENCES cuota(idCuota);

ALTER TABLE colegiatura

ADD FOREIGN KEY (idEstadoSolvencia) REFERENCES
estadoSolvencia(idEstadoSolvencia);

ALTER TABLE colegiatura

ADD FOREIGN KEY (idAlumno) REFERENCES
alumno(idAlumno);

ALTER TABLE planEstudio ADD idColegiatura INT;

ALTER TABLE planEstudio ADD idNivelAcademico INT;

ALTER TABLE planEstudio

ADD FOREIGN KEY (idColegiatura) REFERENCES
colegiatura(idColegiatura);

ALTER TABLE planEstudio

```
ADD FOREIGN KEY (idNivelAcademico) REFERENCES  
nivelAcademico(idNivelAcademico);
```

```
ALTER TABLE planDiario ADD idDirector INT;
```

```
ALTER TABLE planDiario  
ADD FOREIGN KEY (idDirector) REFERENCES  
director(idDirector);
```

```
ALTER TABLE planFinSemana ADD idDirector INT;
```

```
ALTER TABLE planFinSemana  
ADD FOREIGN KEY (idDirector) REFERENCES  
director(idDirector);
```

```
ALTER TABLE planMadurez ADD idDirector INT;
```

```
ALTER TABLE planMadurez  
ADD FOREIGN KEY (idDirector) REFERENCES  
director(idDirector);
```

```
ALTER TABLE jornada ADD idPlanEstudio INT;
```

```
ALTER TABLE jornada
```

ADD FOREIGN KEY (idPlanEstudio) REFERENCES
planEstudio(idPlanEstudio);

ALTER TABLE grado ADD idSeccion INT;

ALTER TABLE grado

ADD FOREIGN KEY (idSeccion) REFERENCES
seccion(idSeccion);

ALTER TABLE curso ADD idBimestre INT;

ALTER TABLE curso ADD idMaestro INT;

ALTER TABLE curso ADD idPensum INT;

ALTER TABLE curso

ADD FOREIGN KEY (idBimestre) REFERENCES
bimestre(idBimestre);

ALTER TABLE curso

ADD FOREIGN KEY (idMaestro) REFERENCES
maestro(idMaestro);

ALTER TABLE curso

ADD FOREIGN KEY (idPensum) REFERENCES
pensum(idPensum);

```
ALTER TABLE pensum ADD idGrado INT;  
  
ALTER TABLE pensum  
  
ADD FOREIGN KEY (idGrado) REFERENCES grado(idGrado);
```

```
ALTER TABLE nivelAcademico ADD idJornada INT;
```

```
ALTER TABLE nivelAcademico ADD idPensum INT;
```

```
ALTER TABLE nivelAcademico
```

```
ADD FOREIGN KEY (idJornada) REFERENCES  
jornada(idJornada);
```

```
ALTER TABLE nivelAcademico
```

```
ADD FOREIGN KEY (idPensum) REFERENCES  
pensum(idPensum);
```

```
ALTER TABLE tarea ADD idCurso INT;
```

```
ALTER TABLE tarea ADD idAlumno INT;
```

```
ALTER TABLE tarea
```

```
ADD FOREIGN KEY (idCurso) REFERENCES curso(idCurso);
```

```
ALTER TABLE tarea
```

```
ADD FOREIGN KEY (idAlumno) REFERENCES  
alumno(idAlumno);
```

```
ALTER TABLE asistencia ADD idCurso INT;  
  
ALTER TABLE asistencia ADD idAlumno INT;  
  
ALTER TABLE asistencia  
ADD FOREIGN KEY (idCurso) REFERENCES curso(idCurso);  
  
ALTER TABLE asistencia  
ADD FOREIGN KEY (idAlumno) REFERENCES  
alumno(idAlumno);  
  
  
  
ALTER TABLE evaluacion ADD idCurso INT;  
  
ALTER TABLE evaluacion ADD idAlumno INT;  
  
ALTER TABLE evaluacion ADD idGrado INT;  
  
ALTER TABLE evaluacion ADD idSeccion INT;  
  
ALTER TABLE evaluacion  
ADD FOREIGN KEY (idCurso) REFERENCES curso(idCurso);  
  
ALTER TABLE evaluacion  
ADD FOREIGN KEY (idAlumno) REFERENCES  
alumno(idAlumno);  
  
ALTER TABLE evaluacion  
ADD FOREIGN KEY (idGrado) REFERENCES grado(idGrado);
```

```
ALTER TABLE evaluacion  
ADD FOREIGN KEY (idSeccion) REFERENCES  
seccion(idSeccion);  
  
ALTER TABLE calificacionBimestral ADD idAsistencia INT;  
ALTER TABLE calificacionBimestral ADD idTarea INT;  
ALTER TABLE calificacionBimestral ADD idEvaluacion INT;  
ALTER TABLE calificacionBimestral ADD idAlumno INT;  
ALTER TABLE calificacionBimestral ADD idMaestro INT;  
ALTER TABLE calificacionBimestral  
ADD FOREIGN KEY (idAsistencia) REFERENCES  
asistencia(idAsistencia);  
ALTER TABLE calificacionBimestral  
ADD FOREIGN KEY (idTarea) REFERENCES tarea(idTarea);  
ALTER TABLE calificacionBimestral  
ADD FOREIGN KEY (idEvaluacion) REFERENCES  
evaluacion(idEvaluacion);  
ALTER TABLE calificacionBimestral  
ADD FOREIGN KEY (idAlumno) REFERENCES  
alumno(idAlumno);  
ALTER TABLE calificacionBimestral
```

```
ADD FOREIGN KEY (idMaestro) REFERENCES  
maestro(idMaestro);
```

```
ALTER TABLE calificacionFinalAnual ADD  
idCalificacionBimestral INT;
```

```
ALTER TABLE calificacionFinalAnual ADD idAlumno INT;
```

```
ALTER TABLE calificacionFinalAnual ADD idMaestro INT;
```

```
ALTER TABLE calificacionFinalAnual
```

```
ADD FOREIGN KEY (idCalificacionBimestral) REFERENCES  
calificacionBimestral(idCalificacionBimestral);
```

```
ALTER TABLE calificacionFinalAnual
```

```
ADD FOREIGN KEY (idAlumno) REFERENCES  
alumno(idAlumno);
```

```
ALTER TABLE calificacionFinalAnual
```

```
ADD FOREIGN KEY (idMaestro) REFERENCES  
maestro(idMaestro);
```

```
insert into alumno values ('Gustavus', 'Maryet',  
'gmaryet0@facebook.com', '10/11/2007', '569 Sutherland  
Parkway');
```

```
insert into alumno values ('Tildy', 'Upsale',  
'tupsale1@businesswire.com', '8/1/2009', '4821 Lighthouse Bay  
Park');
```

```
insert into alumno values ( 'Wynne', 'Bartoszewski',
'wbartoszewski2@samsung.com', '1/1/2003', '1 Walton Park');

insert into alumno values ( 'Lanette', 'Casey',
'lcasey3@behance.net', '10/20/2011', '1 Corscot Lane');

insert into alumno values ( 'Vinnie', 'Stollery',
'vestollery4@nhs.uk', '12/15/2006', '2608 Golf View Terrace');

insert into alumno values ( 'Timmie', 'Kelsell',
'tkelsell5@webnode.com', '10/9/2011', '71735 Dawn Pass');

insert into alumno values ( 'Rosemary', 'Kennet',
'rkennet6@cam.ac.uk', '11/28/2012', '650 Coolidge Court');

insert into alumno values ( 'Flossi', 'Hawlgarth',
'fhawlgarth7@trellian.com', '12/9/2007', '67 Mandrake Junction');

insert into alumno values ( 'Rozalie', 'Blankley',
'rblankley8@newsvine.com', '8/20/2009', '5 5th Hill');

insert into alumno values ( 'Lindy', 'Ollet',
'lollet9@dropbox.com', '4/13/2011', '84177 Stephen Park');

insert into alumno values ( 'Rudolf', 'Burren',
'rburrena@slate.com', '4/13/2008', '20559 Thierer Place');

insert into alumno values ( 'Alice', 'Orteau',
'aorteaub@cpanel.net', '8/22/2006', '6 Sutteridge Place');

insert into alumno values ( 'Eula', 'Spare', 'esparec@usda.gov',
'4/8/2007', '2 Stephen Road');
```

insert into alumno values ('Shirlene', 'Cavendish', 'scavendishd@yandex.ru', '8/2/2007', '23002 Sunfield Crossing');

insert into alumno values ('Danni', 'Kytter', 'dkyttere@npr.org', '10/6/2004', '63982 8th Alley');

insert into alumno values ('Kerby', 'Coggings', 'kcoggingsf@vimeo.com', '10/21/2009', '27345 Bowman Way');

insert into alumno values ('Tanya', 'Tongs', 'ttongsg@nature.com', '5/4/2004', '6 Forster Hill');

insert into alumno values ('Gabrielle', 'Henryson', 'ghenrysonh@nydailynews.com', '9/3/2006', '34 Namekagon Junction');

insert into alumno values ('Tobin', 'Trinbey', 'ttrinbeyi@histats.com', '8/6/2013', '0 Bobwhite Pass');

insert into alumno values ('Patti', 'Keast', 'pkeastj@wsj.com', '2/26/2005', '89 Express Terrace');

insert into alumno values ('Hinda', 'Ionnidis', 'hionnidisk@smh.com.au', '1/30/2008', '2 Golf Course Road');

insert into alumno values ('Dorene', 'Tremoulet', 'dtremouletl@scribd.com', '1/12/2012', '231 Myrtle Circle');

insert into alumno values ('Kleon', 'Mucklo', 'kmucklom@weather.com', '2/27/2005', '8 Southridge Pass');

insert into alumno values ('Ofella', 'Gant', 'ogantn@archive.org', '12/21/2008', '5 Chive Hill');

```
insert into alumno values ('Farand', 'Tregiddo',
'ftregiddoo@rediff.com', '6/15/2013', '6407 Kingsford Drive');
```

```
insert into maestro values ('Nesta', 'Cristofolo',
'ncristofolo0@123-reg.co.uk', '6248105996');
```

```
insert into maestro values ('Bibi', 'Greenhouse',
'bgreenhouse1@themeforest.net', '9106739852');
```

```
insert into maestro values ('Mathian', 'Waind',
'mwaind2@illinois.edu', '6146733718');
```

```
insert into maestro values ('Gussy', 'Normandale',
'gnormandale3@sphinn.com', '2028463052');
```

```
insert into maestro values ('Travis', 'Charlick',
'tcharlick4@lycos.com', '8603608850');
```

```
insert into maestro values ('Torrance', 'Charteris',
'tcharteris5@posterous.com', '3227248790');
```

```
insert into maestro values ('Cymbre', 'Bevington',
'cbevington6@instagram.com', '2177530480');
```

```
insert into maestro values ('Kristan', 'Domone',
'kdomone7@biglobe.ne.jp', '3037521545');
```

```
insert into maestro values ('Jo ann', 'Mellanby',
'jmellanby8.blogspot.com', '6703169145');
```

```
insert into maestro values ('Flor', 'Habershon',
'fhabershon9@va.gov', '5941537020');
```

```
insert into maestro values ('Pandora', 'Lightwing',
'plightwinga@studiopress.com', '7137780191');

insert into maestro values ('Vaughan', 'Braiden',
'vebraidenb@army.mil', '2064403511');

insert into maestro values ('Didi', 'Tidman',
'dtidmanc@marketwatch.com', '3407099891');

insert into maestro values ('Linnea', 'Bakhrushin',
'lbakhrushind@wiley.com', '2151246591');

insert into maestro values ('Gardner', 'Baseley',
'gbaseleye@dell.com', '1685725992');

insert into maestro values ('Binnie', 'Coffelt',
'bcoffeltf@pagesperso-orange.fr', '3803158626');

insert into maestro values ('Willem', 'Bedo',
'wbedog@pagesperso-orange.fr', '6519706497');

insert into maestro values ('Ode', 'Lackeye',
'olackeyeh@samsung.com', '6289503661');

insert into maestro values ('Bartholomew', 'Coulthurst',
'bcoulthursti@last.fm', '8547145626');

insert into maestro values ('Barth', 'Dowson',
'bdowsonj@list-manage.com', '8987525977');

insert into maestro values ('Marcello', 'Jeduch',
'mjeduchk@tripadvisor.com', '7663568414');

insert into maestro values ('Jordana', 'Ber',
'jberl@tmall.com', '9611097555');
```

```
insert into maestro values ('Lem', 'Berryann',  
'lberryannm@lulu.com', '6244546367');
```

```
insert into maestro values ('Gae', 'Copson',  
'gcopsonn@miitbeian.gov.cn', '7081094307');
```

```
insert into maestro values ('Idalina', 'Qusklay',  
'iqusklayo@newyorker.com', '9005562483');
```

```
insert into director values ('Callie', 'Shieber', '24237648234',  
'1741673680', 'cshieber0@cyberchimps.com', '14254 John Wall  
Drive');
```

```
insert into director values ('Joeann', 'Goodrum', '72348234627',  
'7995268878', 'jgoodrum1@bandcamp.com', '6 Valley Edge  
Lane');
```

```
insert into director values ('Jacobo', 'Tanswill', '98443894388',  
'3273294857', 'jtanswill2@xinhuanet.com', '835 Lerdahl Pass');
```

```
insert into tutor values ('Leela', 'Woolforde', '049 Dakota  
Parkway', '7419536194', 'Papa');
```

```
insert into tutor values ('Kiele', 'Ziehms', '6761 Namekagon  
Circle', '8789804958', 'Mama');
```

```
insert into tutor values ('Liza', 'Trevna', '39951 Gulseth Way',  
'2916875221', 'Papa');
```

```
insert into tutor values ('Bax', 'Hartfield', '9 Corscot Drive',  
'2068517387', 'Abuela');
```

insert into tutor values ('Oby', 'Scotney', '99 Anthes Parkway', '8161839569', 'Mama');

insert into tutor values ('Timmie', 'Kollatsch', '320 Johnson Lane', '9641612240', 'Abuela');

insert into tutor values ('Moise', 'Alphege', '18517 Mandrake Street', '6259304749', 'Abuelo');

insert into tutor values ('Adiana', 'Aldritt', '8 Del Mar Crossing', '7481409118', 'Abuela');

insert into tutor values ('Sula', 'Bunce', '75 Miller Junction', '4512773489', 'Abuelo');

insert into tutor values ('Kelcie', 'Stanning', '2 Kings Drive', '1744981669', 'Mama');

insert into tutor values ('Arlen', 'Olive', '37506 Oak Valley Drive', '8917992272', 'Abuelo');

insert into tutor values ('Maxim', 'Dyble', '82221 Village Green Terrace', '2134205776', 'Mama');

insert into tutor values ('Pattie', 'McClinton', '38 Oriole Center', '4151824978', 'Abuelo');

insert into tutor values ('Neilla', 'Pizer', '423 Stephen Alley', '4137960234', 'Mama');

insert into tutor values ('Olivier', 'Tomblin', '6774 Glendale Plaza', '4737009156', 'Papa');

insert into tutor values ('Billie', 'Learoyd', '1 Blue Bill Park Avenue', '1488359599', 'Mama');

insert into tutor values ('Laurice', 'Harkin', '332 Prentice Crossing', '4703761413', 'Abuela');

insert into tutor values ('Winifred', 'Tollerton', '0 Oakridge Plaza', '7338730402', 'Abuela');

insert into tutor values ('Florentia', 'Warwicker', '08 Cottonwood Parkway', '1439826107', 'Mama');

insert into tutor values ('Nolana', 'Cheers', '38092 Rieder Street', '3282563476', 'Papa');

insert into tutor values ('Harlan', 'Rayer', '74434 Browning Alley', '4805669780', 'Abuelo');

insert into tutor values ('Wallis', 'Ragbourn', '360 Spohn Road', '3604722490', 'Abuela');

insert into tutor values ('Wanids', 'Geall', '231 Chinook Center', '5031650895', 'Abuela');

insert into tutor values ('Delcine', 'Wooton', '31 Sheridan Circle', '1429066859', 'Papa');

insert into tutor values ('Stavro', 'Sergison', '6 Grasskamp Point', '4562343208', 'Abuela');

insert into instituto values ('Instituto Mega', '78715555', '1 Corscot Lane');

insert into inscripcion values ('11/27/2021');

```
insert into inscripcion values ('11/9/2021');  
insert into inscripcion values ('11/15/2021');  
insert into inscripcion values ('11/30/2021');  
insert into inscripcion values ('1/17/2022');  
insert into inscripcion values ('1/2/2022');  
insert into inscripcion values ('1/29/2022');  
insert into inscripcion values ('11/4/2021');  
insert into inscripcion values ('11/30/2021');  
insert into inscripcion values ('12/20/2021');  
insert into inscripcion values ('1/2/2022');  
insert into inscripcion values ('12/25/2021');  
insert into inscripcion values ('11/25/2021');  
insert into inscripcion values ('11/16/2021');  
insert into inscripcion values ('1/3/2022');  
insert into inscripcion values ('11/2/2021');  
insert into inscripcion values ('12/13/2021');  
insert into inscripcion values ('12/29/2021');  
insert into inscripcion values ('1/8/2022');  
insert into inscripcion values ('1/29/2022');
```

```
insert into inscripcion values ('11/20/2021');
```

```
insert into inscripcion values ('11/15/2021');
```

```
insert into inscripcion values ('11/22/2021');
```

```
insert into inscripcion values ('12/17/2021');
```

```
insert into inscripcion values ('11/21/2021');
```

```
insert into colegiatura values ('9/23/2022');
```

```
insert into colegiatura values ('1/24/2022');
```

```
insert into colegiatura values ('7/2/2022');
```

```
insert into colegiatura values ('9/11/2022');
```

```
insert into colegiatura values ('2/12/2022');
```

```
insert into colegiatura values ('3/12/2022');
```

```
insert into colegiatura values ('1/29/2022');
```

```
insert into colegiatura values ('7/2/2022');
```

```
insert into colegiatura values ('2/15/2022');
```

```
insert into colegiatura values ('1/11/2022');
```

```
insert into colegiatura values ('8/22/2022');
```

```
insert into colegiatura values ('8/27/2022');
```

```
insert into colegiatura values ('5/4/2022');
```

```
insert into colegiatura values ('6/25/2022');

insert into colegiatura values ('5/28/2022');

insert into colegiatura values ('3/30/2022');

insert into colegiatura values ('7/15/2022');

insert into colegiatura values ('2/20/2022');

insert into colegiatura values ('7/18/2022');

insert into colegiatura values ('9/10/2022');

insert into colegiatura values ('8/9/2022');

insert into colegiatura values ('8/2/2022');

insert into colegiatura values ('9/3/2022');

insert into colegiatura values ('6/26/2022');

insert into colegiatura values ('7/3/2022');

insert into cuota values ('200');

insert into cuota values ('250');

insert into cuota values ('300');

insert into estadoSolvencia values ('Solvente');

insert into estadoSolvencia values ('Insolvente');
```

```
insert into estadoSolvencia values ('Solvente');

insert into estadoSolvencia values ('Solvente');

insert into estadoSolvencia values ('Insolvente');

insert into estadoSolvencia values ('Solvente');

insert into estadoSolvencia values ('Solvente');

insert into estadoSolvencia values ('Insolvente');

insert into estadoSolvencia values ('Solvente');

insert into estadoSolvencia values ('Solvente');

insert into estadoSolvencia values ('Insolvente');

insert into estadoSolvencia values ('Solvente');

insert into estadoSolvencia values ('Insolvente');

insert into estadoSolvencia values ('Solvente');

insert into estadoSolvencia values ('Solvente');

insert into estadoSolvencia values ('Insolvente');

insert into estadoSolvencia values ('Solvente');

insert into estadoSolvencia values ('Solvente');

insert into estadoSolvencia values ('Insolvente');

insert into estadoSolvencia values ('Solvente');

insert into estadoSolvencia values ('Solvente');

insert into estadoSolvencia values ('Insolvente');

insert into estadoSolvencia values ('Solvente');

insert into estadoSolvencia values ('Solvente');
```

```
insert into estadoSolvencia values ('Solvente');  
insert into estadoSolvencia values ('Insolvente');  
insert into estadoSolvencia values ('Solvente');  
insert into estadoSolvencia values ('Solvente');
```

```
insert into planEstudio values ('Plan Diario');  
insert into planEstudio values ('Plan Fin de Semana');  
insert into planEstudio values ('Plan por Madurez');
```

```
insert into jornada values ('Jornada Matutina');  
insert into jornada values ('Jornada Vespertina');  
insert into jornada values ('Jornada Nocturna');
```

```
insert into grado values ('Primero Primaria');  
insert into grado values ('Segundo Primaria');  
insert into grado values ('Tercero Primaria');  
insert into grado values ('Cuarto Primaria');  
insert into grado values ('Quinto Primaria');  
insert into grado values ('Sexto Primaria');
```

```
insert into grado values ('Primero Basico');

insert into grado values ('Segundo Basico');

insert into grado values ('Tercero Basico');

insert into grado values ('Cuarto Bachillerato');

insert into grado values ('Quinto Bachillerato');

insert into seccion values ('A');

insert into seccion values ('B');

insert into seccion values ('C');

insert into curso values ('Matematicas, Ciencias Sociales,
Ciencias Naturales, Computacion, Expresion Artistica,
Comunicacion y Lenguaje');

insert into curso values ('Matematicas, Ciencias Sociales,
Ciencias Naturales, Computacion, Expresion Artistica,
Comunicacion y Lenguaje, Contabilidad');

insert into curso values ('Matematicas, Ciencias Sociales,
Ciencias Naturales, Computacion, Expresion Artistica,
Comunicacion y Lenguaje, Contabilidad');

insert into curso values ('Matematicas, Ciencias Sociales,
Ciencias Naturales, Computacion, Expresion Artistica,
Comunicacion y Lenguaje, Contabilidad, Seminario');
```

insert into bimestre values ('Primer Bimestre');

insert into bimestre values ('Segundo Bimestre');

insert into bimestre values ('Tercer Bimestre');

insert into bimestre values ('Cuarto Bimestre');

insert into pensum values ('Pensum Primaria');

insert into pensum values ('Pensum Basico');

insert into pensum values ('Pensum Cuarto Diversificado');

insert into pensum values ('Pensum Quinto Diversificado');

insert into genero values ('F');

insert into genero values ('M');

insert into genero values ('M');

insert into genero values ('F');

insert into genero values ('M');

```
insert into genero values ('F');

insert into genero values ('M');

insert into genero values ('M');

insert into genero values ('F');

insert into asignaturaExtra values ('IDIOMA EXTRANJERO');
```

```
insert into asignaturaExtra values ('IDIOMA MAYA');

insert into asignaturaExtra values ('IDIOMA EXTRANJERO');

insert into asignaturaExtra values ('IDIOMA MAYA');

insert into asignaturaExtra values ('IDIOMA EXTRANJERO');

insert into asignaturaExtra values ('IDIOMA MAYA');

insert into asignaturaExtra values ('IDIOMA EXTRANJERO');

insert into asignaturaExtra values ('IDIOMA EXTRANJERO');

insert into asignaturaExtra values ('IDIOMA MAYA');

insert into asignaturaExtra values ('IDIOMA EXTRANJERO');

insert into asignaturaExtra values ('IDIOMA MAYA');

insert into asignaturaExtra values ('IDIOMA EXTRANJERO');

insert into asignaturaExtra values ('IDIOMA MAYA');

insert into asignaturaExtra values ('IDIOMA EXTRANJERO');
```

```
insert into asignaturaExtra values ('IDIOMA MAYA');

insert into asignaturaExtra values ('IDIOMA MAYA');

insert into asignaturaExtra values ('IDIOMA EXTRANJERO');

insert into asignaturaExtra values ('IDIOMA MAYA');

insert into asignaturaExtra values ('IDIOMA MAYA');

insert into tarea values (58, '10/5/2022');

insert into tarea values (52, '10/7/2022');

insert into tarea values (67, '10/9/2022');

insert into tarea values (59, '10/8/2022');

insert into tarea values (63, '10/6/2022');

insert into tarea values (58, '10/9/2022');

insert into tarea values (56, '10/7/2022');

insert into tarea values (64, '10/8/2022');

insert into tarea values (53, '10/6/2022');

insert into tarea values (63, '10/7/2022');

insert into tarea values (60, '10/6/2022');

insert into tarea values (54, '10/8/2022');

insert into tarea values (58, '10/7/2022');
```

```
insert into tarea values (69, '10/6/2022');  
insert into tarea values (73, '10/5/2022');  
insert into tarea values (62, '10/6/2022');  
insert into tarea values (50, '10/9/2022');  
insert into tarea values (51, '10/6/2022');  
insert into tarea values (53, '10/8/2022');  
insert into tarea values (87, '10/6/2022');  
insert into tarea values (71, '10/8/2022');  
insert into tarea values (81, '10/6/2022');  
insert into tarea values (97, '10/7/2022');  
insert into tarea values (76, '10/6/2022');  
insert into tarea values (96, '10/5/2022');
```

```
insert into asistencia values (9);  
insert into asistencia values (9);  
insert into asistencia values (2);  
insert into asistencia values (7);  
insert into asistencia values (6);  
insert into asistencia values (7);
```

```
insert into asistencia values (5);  
insert into asistencia values (6);  
insert into asistencia values (5);  
insert into asistencia values (4);  
insert into asistencia values (7);  
insert into asistencia values (8);  
insert into asistencia values (2);  
insert into asistencia values (8);  
insert into asistencia values (6);  
insert into asistencia values (3);  
insert into asistencia values (5);  
insert into asistencia values (8);  
insert into asistencia values (6);  
insert into asistencia values (4);  
insert into asistencia values (7);  
insert into asistencia values (8);
```

```
insert into evaluacion values ('10/1/2022', 28);
insert into evaluacion values ('10/1/2022', 13);
insert into evaluacion values ('10/1/2022', 38);
insert into evaluacion values ('10/1/2022', 11);
insert into evaluacion values ('10/1/2022', 41);
insert into evaluacion values ('10/1/2022', 4);
insert into evaluacion values ('10/1/2022', 46);
insert into evaluacion values ('10/1/2022', 6);
insert into evaluacion values ('10/1/2022', 22);
insert into evaluacion values ('10/1/2022', 13);
insert into evaluacion values ('10/1/2022', 41);
insert into evaluacion values ('10/1/2022', 45);
insert into evaluacion values ('10/1/2022', 7);
insert into evaluacion values ('10/1/2022', 48);
insert into evaluacion values ('10/1/2022', 3);
insert into evaluacion values ('10/1/2022', 8);
insert into evaluacion values ('10/1/2022', 8);
insert into evaluacion values ('10/1/2022', 16);
```

```
insert into evaluacion values ('10/1/2022', 25);
insert into evaluacion values ('10/1/2022', 9);
insert into evaluacion values ('10/1/2022', 40);
insert into evaluacion values ('10/1/2022', 47);
insert into evaluacion values ('10/1/2022', 2);
insert into evaluacion values ('10/1/2022', 14);
insert into evaluacion values ('10/1/2022', 11);
```

```
insert into calificacionBimestral values (37);
insert into calificacionBimestral values (2);
insert into calificacionBimestral values (60);
insert into calificacionBimestral values (8);
insert into calificacionBimestral values (51);
insert into calificacionBimestral values (50);
insert into calificacionBimestral values (99);
insert into calificacionBimestral values (68);
insert into calificacionBimestral values (7);
insert into calificacionBimestral values (55);
insert into calificacionBimestral values (1);
```

```
insert into calificacionBimestral values (1);  
insert into calificacionBimestral values (94);  
insert into calificacionBimestral values (63);  
insert into calificacionBimestral values (89);  
insert into calificacionBimestral values (40);  
insert into calificacionBimestral values (60);  
insert into calificacionBimestral values (45);  
insert into calificacionBimestral values (1);  
insert into calificacionBimestral values (38);  
insert into calificacionBimestral values (2);  
insert into calificacionBimestral values (93);  
insert into calificacionBimestral values (1);  
insert into calificacionBimestral values (82);  
insert into calificacionBimestral values (16);  
  
insert into calificacionFinalAnual values (66);  
insert into calificacionFinalAnual values (45);  
insert into calificacionFinalAnual values (56);  
insert into calificacionFinalAnual values (68);
```

```
insert into calificacionFinalAnual values (44);
insert into calificacionFinalAnual values (70);
insert into calificacionFinalAnual values (47);
insert into calificacionFinalAnual values (41);
insert into calificacionFinalAnual values (63);
insert into calificacionFinalAnual values (97);
insert into calificacionFinalAnual values (61);
insert into calificacionFinalAnual values (50);
insert into calificacionFinalAnual values (98);
insert into calificacionFinalAnual values (65);
insert into calificacionFinalAnual values (90);
insert into calificacionFinalAnual values (82);
insert into calificacionFinalAnual values (64);
insert into calificacionFinalAnual values (70);
insert into calificacionFinalAnual values (49);
insert into calificacionFinalAnual values (100);
insert into calificacionFinalAnual values (78);
insert into calificacionFinalAnual values (82);
insert into calificacionFinalAnual values (69);
```

```
insert into calificacionFinalAnual values (100);
```

```
insert into calificacionFinalAnual values (99);
```

-- CONSULTAS

```
select*from alumno
```

```
select*from maestro
```

```
select*from director
```

```
select*from tutor
```

```
select*from instituto
```

```
select*from inscripcion
```

```
select*from colegiatura
```

```
select*from cuota
```

```
select*from estadoSolvencia
```

```
select*from planEstudio
```

```
select*from planDiario
```

```
select*from planFinsemana
```

```
select*from planMadurez
```

```
select*from jornada
```

```
select*from grado  
select*from seccion  
select*from curso  
select*from bimestre  
select*from pensum  
select*from nivelAcademico  
select*from genero  
select*from asignaturaExtra  
select*from tarea  
select*from asistencia  
select*from evaluacion  
select*from calificacionBimestral  
select*from calificacionFinalAnual
```

-- MÁS CONSULTAS

--#1 Realiza la consulta en la tabla Evaluacion con las notas > 70 y < 60

```
SELECT * FROM evaluacion WHERE notaEvaluacion > 0 AND  
notaEvaluacion < 60;
```

--#2 Realiza la consulta en la tabla Evaluacion entre el idAlumno de 1 al 5 con una notaEvaluacion de 40 pts.

```
SELECT * FROM evaluacion WHERE (idEvaluacion > 1 AND notaEvaluacion < 5) OR notaEvaluacion = 40;
```

--#3 Realiza la consulta en la tabla estadoSolvente sobre quienes no están Solvente

```
SELECT * FROM estadoSolvencia WHERE NOT Estado = 'Solvente';
```

--#4 selecciona todos los grados que se encuentran en "Tercero Basico", "Tercero Primaria"

```
SELECT * FROM grado WHERE nombreGrado IN ('Tercero Primaria', 'Tercero Basico');
```

--#5 La siguiente instrucción selecciona todos los alumnos con un nombre que comienza con "a"

```
SELECT * FROM alumno WHERE nombreAlumno LIKE 'a%';
```

--#6 La siguiente instrucción selecciona todos los tutor con la direccion "049 Dakota Parkway", en la tabla "direccionTutor"

```
SELECT * FROM tutor WHERE direccionTutor='049 Dakota Parkway';
```

--#7 --total de personas por genero

```
select descripcion , count(*) as genero from genero  
group by descripcion
```

--#8

```
SELECT idAsistencia, notaAsistencia FROM asistencia WHERE  
notaAsistencia <= 5;
```

--#9 La siguiente instrucción enumera la cantidad de integrantes de planEstudio

```
SELECT COUNT(DISTINCT descripcion) FROM planEstudio;
```

--#10 La siguiente instrucción SQL selecciona todos los clientes de la tabla "Clientes", ordenados de forma ascendente por el "País" y descendente por la columna "Nombre del cliente":

```
SELECT * FROM alumno ORDER BY nacimientoAlumno ASC,  
direccionAlumno DESC;
```

--#11 descendente

```
select nombreAlumno,apellidoAlumno,correoAlumno from  
alumno order by correoAlumno desc
```

--#12 ascendente

```
select nombreAlumno,apellidoAlumno,correoAlumno from  
alumno order by correoAlumno asc
```

--#13 Estudiantes que tienen asignaturaExtra de IDIOMA MAYA

```
select * from asignaturaExtra where idAsignaturaExtra in(select  
idAsignaturaExtra from asignaturaExtra where  
nombreAsignatura='IDIOMA MAYA')
```

--#14 Realiza la consulta de la cuota mas elevada

```
select * from cuota where montoCuota = (select MAX  
(montoCuota) from cuota)
```

--#15 Cuantas calificaciones existen entre 80 y 100 pts en la
tabla calificacionFinalAnual

```
select * from calificacionFinalAnual where notaFinalAnual  
between 80 and 100;
```

--SUBCONSULTAS

--#1 Lo que se esta haciendo es desde los alumnos se obtienen en los nombres de los alumnos del menor al promedio de la jornada

```
select nombreAlumno, idGrado from alumno where  
gradoAlumno <(select avg(jornada) from jornadaAlumno;
```

--#2 Lo que se esta haciendo es desde los alumnos se obtienen el los nombres de los alumnos mayor al promedio de la jornada

```
SELECT * from alumno where idAlumno >(select AVG(idGrado)  
from grado);
```

--#3 Se selecciona todos los alumnos que esten asignados en el curso de matematica

```
SELECT nombreAlumno from alumno where nombreAlumno >  
all (select idCurso from curso where nombreCurso  
='matematica');
```

--#4 Se selecciona algunos alumnos que esten asignados en el curso de matematica

```
select * from curso where nombreCurso > any (select idCurso  
from alumno LIKE 'matematica%');
```

--#5 Se mostrara los alumnos que tengan una asistencia de 20 veces

```
select nombreAlumno from alumno) where asistencia in (select  
grado from asistencia where asistencia >20);
```

--JOINS

--#1

```
select * from evaluaciones eval  
full outerjoin maestro mae  
on eval.evaluaiones = mae.idevaluacion
```

--#2

```
select jordaNocturna  
from jornadas jor  
full outer join planDiario pld  
on jor.idjornadas = pld.idplanDiarioN  
where jor.idjornadas is null  
or pld.idplanDiarioN is null
```

--#3

```
select nombreAlumno  
from inscripcion ins  
right join gradoencurso gra  
on ins.idinscripcion = gra.idgrado
```

--#4

```
select *  
from seccion secn  
right join nombreAlumno alm  
on secn.idseccion = alm.idAlumno  
where secn.idseccion is null
```

--#5

```
select *
```

```
from seccion secn  
right join nombreAlumno alm  
on secn.idseccion = alm.idAlumno  
where alm.idAlumno is null
```

```
SELECT alumno.idAlumno, alumno.nombreAlumno,  
maestro.nombreMaestro FROM alumno INNER JOIN maestro ON  
alumno.idAlumno = maestro.idMaestro
```

--procedimiento que almacene los datos del alumno y sus notas bimestrales

USE notas

GO

```
CREATE PROCEDURE nombreXnotas
```

```
@busca varchar(15)
```

AS

BEGIN

```
SELECT grado.idGrado, curso.idCurso,  
curso.calificacionBimestral
```

```
FROM grado INNER JOIN calificacionBimestral ON  
calificacionBimestral.idcalificacionBimestral = curso.idcurso
```

WHERE grado.grado like @busca

END