



BASE DE DATOS – SQL SERVER

ESTUDIANTE:

QUIROGA HUARISTE ANDRES VLADIMIR

DOCENTE:

BARRA PAREDES WILLIAM RODDY

CARRERA:

INGENIERIA DE SISTEMAS

Consigna

Diseñe un sistema de Base de Datos Relacional utilizando el gestor de Base de Datos **SQL Server** teniendo como premisa el uso de buenas prácticas en diseño de la base de datos aplicados al siguiente escenario.

Una comunidad de estudiantes de la nación **UNIFRANZ** de nombre los **UNIFRANZITOS** desea implementar un nuevo sistema para poder administrar los **CAMPEONATOS DE FÚTBOL** de todas las sedes.

Es decir crear un campeonato en donde puedan participar todas las sedes, en el campeonato pueden inscribirse tanto categoría varones y mujeres.

Detalle del problema

UNIFRANZITOS	
Problema	<p>Se tiene como contexto un CAMPEONATO DE FÚTBOL en el cual se tiene 3 entidades principales el campeonato como tal, los equipos que participaran en el campeonato y en donde cada equipo tendrá una cantidad de jugadores.</p> <p>En tal sentido se deberá crear las siguientes tablas.</p> <ul style="list-style-type: none">• campeonato• equipo• jugador <p>Detalle de las tablas.</p>

campeonato

id_campeonato => cadena de 12 caracteres y ademas llave primaria

nombre_campeonato => una cadena de 30 caracteres que no acepta valores nulos

sede => una cadena de 20 caracteres que no acepta valores nulos

equipo

id_equipo => cadena de 12 caracteres y ademas llave primaria

nombre_equipo => una cadena de 30 caracteres, que no acepta valores nulos

categoria => esta columna recibe valores como (varones o mujeres), que no acepta valores nulos

id_campeonato => llave foreign key relacionado con la tabla **campeonato**

jugador

id_jugador => cadena de 12 caracteres y ademas llave primaria

nombres => una cadena de 30 caracteres, que no acepta valores nulos

apellidos => una cadena de 50 caracteres, que no acepta valores nulos

ci => una cadena de 15 caracteres (ejem: 8997899LP), que no acepta valores nulos

edad => un valor numérico, que no acepta valores nulos

id_equipo => llave foreign key relacionado con la tabla **equipo**

1. Diseño de base de datos.

1.1. Dado el detalle explicado en la parte inicial de este documento debería generar una base de datos similar al siguiente.



1.2. Los registros de cada tabla deberían quedar de la siguiente forma

tabla campeonato


 id_campeonato	nombre_campeonato	sede
camp-111	Campeonato Unifranz	El Alto
camp-222	Campeonato Unifranz	Cochabamba

tabla equipo





 id_equipo	nombre_equipo	categoria	 id_campeonato
equ-111	Google	VARONES	camp-111
equ-222	404 Not found	VARONES	camp-111
equ-333	girls unifranz	MUJERES	camp-111

tabla jugador

 id_jugador	nombres	apellidos	ci	edad	 id_equipo
jug-111	Carlos	Villa	8997811LP	19	equ-222
jug-222	Pedro	Salas	8997822LP	20	equ-222
jug-333	Saul	Araj	8997833LP	21	equ-222
jug-444	Sandra	Solis	8997844LP	20	equ-333
jug-555	Ana	Mica	8997855LP	23	equ-333

■ 2 Manejo de Conceptos

2.1 Adjuntar el diagrama E-R Generado por su editor (DATAGRIP o SQL SERVER MANAGERMENTS STUDIO)



- 2.2 Que es DDL y muestre un ejemplo UNIFRANZITOS

Es un lenguaje de definicion de datos es un subconjunto de SQL.

```
create database UNIFRANZITOS
use UNIFRANZITOS

DROP DATABASE UNIFRANZITOS

create table UNIFRANZITOS
(
    id_UNIFRANZITOS varchar(50) PRIMARY KEY ,
    nombres varchar(50),
    edad integer,
    sede varchar(50),
);
ALTER TABLE UNIFRANZITOS
add edad varchar(12)
```

- 2.3 Que es DML y muestre un ejemplo aplicado a la base de datos UNIFRANZITOS

Es un idioma proporcionado por los sistemas gestores de bases de datos que permite a los usuarios de la misma llevar a cabo las tareas de consulta o modificación de los datos contenidos en las Bases de Datos del Sistema Gestor de Bases de Datos.

```
✓ select UNI.*  
from UNIFRANZITOS as UNI;
```

UNIFRANZITOS

Output UNIFRANZITOS x

0 rows

id_UNIFRANZITOS nombres edad sede

2.4 Que significa PRIMARY KEY y FOREIGN KEY.

Una primary key es una columna de tabla que sirve a un propósito especial. Cada tabla necesita una primary key porque garantiza la accesibilidad a nivel de fila (registro).

En el contexto de bases de datos relacionales, una clave foránea o llave foránea o clave ajena (o **Foreign Key FK**) es una limitación referencial entre dos tablas.

- 2.5 Defina que es una TABLA y que es una VISTA

Las tablas son objetos de base de datos que contienen todos sus datos.

Una vista es solo una instrucción SELECT que se ha guardado en la base de datos.

- 2.6 Como funciona LIKE en una consulta SQL. Adjunte un ejemplo.

LIKE en SQL se usa junto a la cláusula WHERE.

```
select jugador.*  
from jugador as jugador  
where jugador.apellidos like 'S%';
```

- **2.7 Para que se utiliza la clausula WHERE.**

Especifica criterios que tienen que cumplir los valores de campo para que los registros que contienen los valores se incluyan en los resultados de la consulta.

- **2.8. Apoyándonos en el concepto de conjuntos muestre los siguiente:**

2.8.1. Ejemplo de INNER JOIN

2.8.2. Adjuntar una imagen de conjuntos y la consulta SQL que refleje el INNER JOIN

```
select equi.nombre_equipo, jug.nombres, jug.id_jugador
from jugador as jug
inner join equipo as equi 1..n<->1: on jug.id_equipo = equi.id_equipo
where id_jugador = 'jug-333';
```

- ▶ 2.9. Apoyándonos en el concepto de conjuntos muestre los siguiente:
- ▶ 2.9.1. Ejemplo de LEFT JOIN
- ▶ 2.9.2. Adjuntar una imagen de conjuntos y la consulta SQL que refleje el LEFT JOIN

```
select equi.id_campeonato, equi.categoria, equi.nombre_equipo
from equipo as equi
left join campeonato as cam 1..n<->0..1: on equi.id_campeonato = cam.id_campeonato
where equi.categoria = 'Mujeres';
```

- ▶ 2.10. Apoyándonos en el concepto de conjuntos muestre los siguiente:
- ▶ 2.10.1. Ejemplo de RIGHT JOIN
- ▶ 2.10.2. Adjuntar una imagen de conjuntos y la consulta SQL que refleje el RIGHT JOIN

```
select equi.id_campeonato, equi.categoria, equi.nombre_equipo
from equipo as equi
right join campeonato as cam 0..n<->1: on equi.id_campeonato = cam.id_campeonato
where equi.categoria = 'Mujeres';
```

• 3 Manejo de Consultas

- 3.1. Mostrar que jugadores que formen parte del equipo equ-222

```
--Mostrar jugadores que formen parte del equipo equ-222
✓ select jugador.*
  from jugador as jugador
 where jugador.id_equipo = 'equ-222';
```

Output Mostrar jugadores qu...te del equipo equ-222 hito3.dbo.equipo 2

3 rows

	id_jugador	id_equipo	nombres	apellidos	ci	edad
1	jug-111	equ-222	Carlos	Villa	8997811LP	19
2	jug-222	equ-222	Pedro	Salas	8997822LP	20
3	jug-333	equ-222	Saul	Araj	8997833LP	21

► 3.2. Mostrar que jugadores que formen parte del equipo equ-333

```
--Mostrar jugadores que formen parte del equipo equ-333
✓ select jugador.*
  from jugador as jugador
 where jugador.id_equipo = 'equ-333';
```

Output Mostrar jugadores qu...te del equipo equ-222 Mostrar jugadores qu...te del equipo equ-333

2 rows

	id_jugador	id_equipo	nombres	apellidos	ci	edad
1	jug-444	equ-333	Sandra	Solis	8997844LP	20
2	jug-555	equ-333	Ana	Mica	8997855LP	23

- 3.3. Mostrar aquellos jugadores mayores o igual a 21 años

```
--Mostrar aquellos jugadores mayores o igual a 21 años  
✓ select jugador.*  
  from jugador as jugador  
 where jugador.edad >=21;
```

Output

Mostrar aquellos jug...es o igual a 21 años X

Mostrar jugadores qu...te del equipo equ-333 X

|< < 2 rows v > >|



Tx: Auto v

DDL



! id_jugador

! id_equipo

! nombres

! apellidos

! ci

! edad

1 jug-333

equ-222

Saul

Araj

8997833LP

21

2 jug-555

equ-333

Ana

Mica

8997855LP

23

- 3.4. Mostrar a todos los estudiantes en donde su apellido empiece con la letra S.

```
--Mostrar a todos los estudiantes en donde su apellido empiece con la letra S  
✓ select jugador.*  
  from jugador as jugador  
 where jugador.apellidos like 'S%';
```

Output Mostrar aquellos jug...es o igual a 21 años × Mostrar a todos los ...mpiece con la letra S ×

	id_jugador	id_equipo	nombres	apellidos	ci	edad
1	jug-222	equ-222	Pedro	Salas	8997822LP	20
2	jug-444	equ-333	Sandra	Solis	8997844LP	20

- 3.5. Mostrar que equipos forman parte del campeonato camp-111 y además sean de la categoría MUJERES.

```
--Mostrar que equipos forman parte del campeonato camp-111 y ademas sean de la categoria Mujeres
✓ select equi.id_campeonato, equi.categoria, equi.nombre_equipo
  from equipo as equi
 inner join campeonato as cam 1..n<->1: on equi.id_campeonato = cam.id_campeonato
 where equi.categoria = 'Mujeres' and cam.id_campeonato = 'camp-111';
```

Output Mostrar que equipos ... la categoria Mujeres × Mostrar a todos los ...mpiece con la letra S ×

1 row Tx: Auto DDL

	id_campeonato	categoria	nombre_equipo
1	camp-111	Mujeres	girls unifranz

- 3.6. Mostrar el nombre del equipo del jugador con id_jugador igual a jug-333

```
--Mostrar el nombre del equipo del jugador con id_jugador igual a jug-333
✓ select equi.nombre_equipo, jug.nombres, jug.id_jugador
  from jugador as jug
 inner join equipo as equi 1..n<->1: on jug.id_equipo = equi.id_equipo
 where id_jugador = 'jug-333';
```

equi

Output				Mostrar que equipos ... la categoria Mujeres		Mostrar el nombre de...gador igual a ju	
1 row							
	nombre_equipo	nombres	id_jugador				
1	404 not found	Saul	jug-333				

3.7. Mostrar el nombre del campeonato del jugador con id_jugador igual

```
--Mostrar el nombre del campeonato del jugador con id_jugador igual a jug-333  
✓ select cam.nombre_campeonato , jug.id_jugador  
   from jugador as jug  
   inner join equipo as equi 1..n<->1: on jug.id_equipo = equi.id_equipo  
   inner join campeonato as cam 1..n<->1: on equi.id_campeonato = cam.id_campeonato  
   where jug.id_jugador = 'jug-333';
```

Output

Mostrar el nombre de...gador igual a jug-333 2

Mostrar el nombre de...gador igual a jug-333

1 row

	nombre_campeonato	id_jugador
1	Campeonato Unifranz	jug-333

- 3.8. Crear una consulta SQL que maneje las 3 tablas de la base de datos.

```
--Crear una consulta SQL que maneje las 3 tablas de la base de datos
--mostrar la sede donde estan jugando y mostrar el id_jugador = jug-222
✓ select cam.sede , jug.id_jugador
from jugador as jug
inner join equipo as equi 1..n<->1: on jug.id_equipo = equi.id_equipo
inner join campeonato as cam 1..n<->1: on equi.id_campeonato = cam.id_campeonato
where jug.id_jugador = 'jug-222';
```

Output Mostrar el nombre de...gador igual a jug-333 2 mostrar la sede dond... id_jugador

|< < 1 row v > >| ↺ ■ ★

■ sede ⇅ ■ id_jugador ⇅

1	El Alto	jug-222
---	---------	---------

3.9. ¿Qué estrategia utilizaría para determinar cuántos equipos inscritos hay?

```
--¿Qué estrategia utilizaría para determinar cuántos equipos inscritos hay?  
✓ select Count (id_equipo)  
   from equipo as equi;
```

Output Count (id_equipo):int mostrar la sede dond... id_jugador = jug-222

1 row

<anonymous>

1	3
---	---

- 3.10. ¿Qué estrategia utilizaría para determinar cuántos jugadores pertenecen a la categoría VARONES o Categoría MUJERES.
- 3.10.1. Para esto puede utilizar la función de agregación COUNT


```
--¿Qué estrategia utilizaría para determinar cuántos jugadores pertenecen a la categoría VARONES o Categoría mujeres
select Count(*) as cantidad_Varones
from equipo as equi
where equi.categoria = 'Varones';


select Count(*) as cantidad_Mujeres
from equipo as equi
where equi.categoria = 'Mujeres';
```




Output cantidad_Mujeres:int

cantidad_Mujeres	
1	1

--¿Qué estrategia utilizaría para determinar cuántos jugadores pertenecen a la categoría VARONES o Categoría mujeres


✓  `select Count(*) as cantidad_Varones
from equipo as equi
where equi.categoria = 'Varones';`

 `select Count(*) as cantidad_Mujeres
from equipo as equi
where equi.categoria = 'Mujeres';`

 Output  cantidad_Varones:int 

|< < 1 row ▾ > >|   

CSV ▾

 cantidad_Varones ▾

1

2