



BASE DE DATOS

Hito 3

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.

```
create database UNIFRANZITOS;

use UNIFRANZITOS;

create table Campeonato(
    id_campeonato varchar(12) primary key,
    nombre_campeonato varchar(30) not null,
    sede varchar(20) not null,
);

create table Equipo (
    id_equipo varchar(12) primary key,
    nombre_equipo varchar(30) not null,
    categoria varchar(8) not null,
    id_campeonato varchar(12),
    foreign key (id_campeonato) references campeonato (id_campeonato),
);

create table Jugador (
    id_jugador varchar(12) primary key,
    nombres varchar(30) not null,
    apellidos varchar(50) not null,
    ci varchar(15) not null,
    edad int not null,
    id_equipo varchar(12),
    foreign key (id_equipo) references equipo(id_equipo),
);
```

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

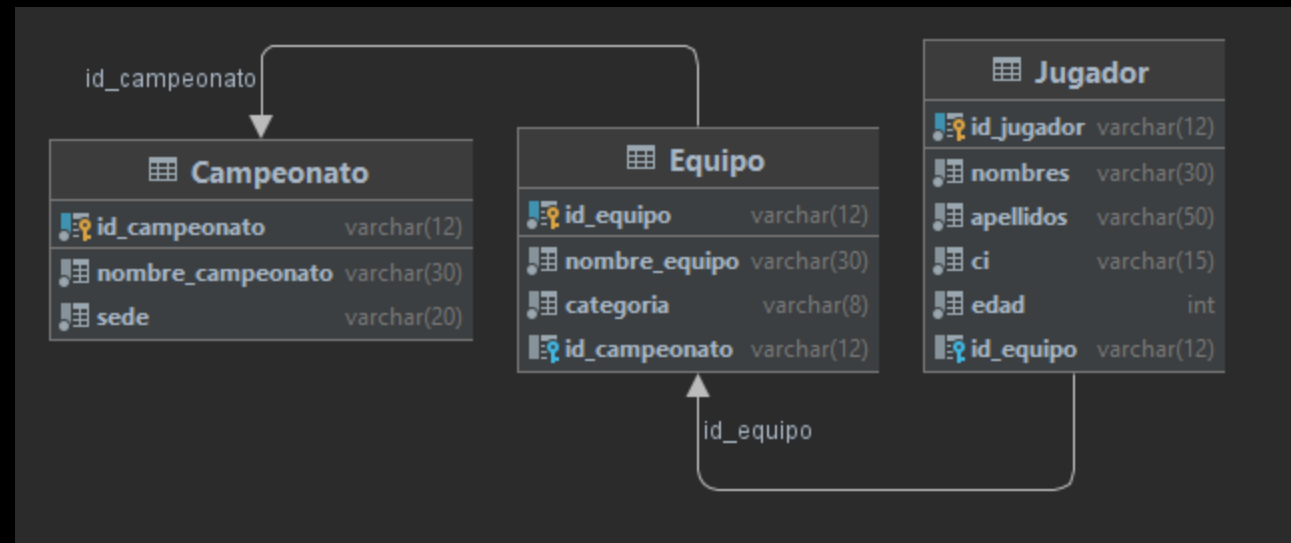
```
insert into Campeonato(id_campeonato, nombre_campeonato, sede)
values('camp-111', 'Campeonato Unifranz', 'El Alto'),
      ('camp-222', 'Campeonato Unifranz', 'Cochabamba');

insert into Equipo(id_equipo, nombre_equipo, categoria, id_campeonato)
values('equ-111', 'Google', 'VARONES', 'camp-111'),
      ('equ-222', '404 not found', 'VARONES', 'camp-111'),
      ('equ-333', 'girls unifranz', 'MUJERES', 'camp-111');

insert into Jugador(id_jugador, nombres, apellidos, ci, edad, id_equipo)
values('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 MANAGEMENTS STUDIO)



2.2. Que es DDL y muestre un ejemplo UNIFRANZITOS.

```
create table Campeonato(  
    id_campeonato varchar(12) primary key,  
    nombre_campeonato varchar(30) not null,  
    sede varchar(20) not null,  
);
```

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

```
insert into Campeonato(id_campeonato, nombre_campeonato, sede)  
values('camp-111', 'Campeonato Unifranz', 'El Alto'),  
      ('camp-222', 'Campeonato Unifranz', 'Cochabamba');
```

2.4. Que significa PRIMARY KEY y FOREIGN KEY.

Las columnas que forman la clave principal no pueden contener valores nulos ni pueden haber valores duplicados de la combinación de columnas. La cláusula **FOREIGN KEY** (Llave foránea) sirve para definir una clave foránea sobre una columna o una combinación de columnas.

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

Las **tablas** son los objetos de base de datos que se comportan como contenedores de información, en los cuales la información será lógicamente organizada en formato de filas y columnas.

Una **vista** es una **tabla** virtual cuyo contenido está definido por una consulta.

2.6. Cómo funciona LIKE en una consulta SQL. Adjunte un ejemplo

El “like” funciona para buscar una determinada palabra o letra.

Mostrar a todos los jugadores que su nombre empiece con una “s”.

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

2.7. Para que se utiliza la cláusula WHERE.

Se utiliza en una consulta y su función es la de poner una condición de búsqueda.

Para los siguientes ejercicios crear 2 tablas cualesquiera.

```
create table Colegio(  
    id_colegio varchar(30) primary key,  
    nombre_colegio varchar(35),  
    nombre_maestros varchar(30),  
    materias varchar(40),  
);  
  
create table Estudiante(  
    id_Estudiante int primary key,  
    nombre varchar(25),  
    apellido varchar(25),  
    edad int,  
    id_colegio varchar(30),  
    foreign key (id_colegio) references Colegio(id_colegio),  
);
```

```
insert into Colegio(id_colegio , nombre_colegio , nombre_maestros, materias)  
values('cole-156', '31 de octubre', 'Jose', 'fis-111'),  
('cole-354', 'illimani 26 de abril', 'Carlos', 'qui-222'),  
('cole-896', 'italia', 'Rodrigo', 'alg-333'),  
('cole-456', 'san antonio', 'Juan', 'pro-444'),  
('cole-789', 'jardines del sur', 'Raul', 'cal-555');  
  
insert into Estudiante (id_Estudiante , nombre, apellido, edad, id_colegio)  
values(1, 'Carlos', 'Villa', 19, 'cole-156'),  
(2, 'Pedro', 'Salas', 20, 'cole-354'),  
(3, 'Saul', 'Araji', 21, 'cole-789'),  
(4, 'Sandra', 'Solis', 20, 'cole-456'),  
(5, 'Ana', 'Mica', 23, 'cole-156');
```

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

2.8.1. Ejemplo de INNER JOIN

```
select est.nombre, col.nombre_colegio  
from Estudiante as est  
    inner join Colegio as col on est.id_colegio = col.id_colegio;
```

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

	nombre	nombre_colegio
1	Carlos	31 de octubre
2	Pedro	illimani 26 de abril
3	Saul	jardines del sur
4	Sandra	san antonio
5	Ana	31 de octubre

2.9. Apoyándonos en el concepto de conjuntos muestre los siguiente:

2.9.1. Ejemplo de LEFT JOIN

```
select est.nombre, col.nombre_colegio  
from Estudiante as est  
left join Colegio as col on est.id_colegio = col.id_colegio;
```

2.9.2. Adjuntar una imagen de conjuntos y la consulta SQL que refleje el LEFT JOIN

	nombre	nombre_colegio
1	Carlos	31 de octubre
2	Pedro	illimani 26 de abril
3	Saul	jardines del sur
4	Sandra	san antonio
5	Ana	31 de octubre

2.10. Apoyándonos en el concepto de conjuntos muestre los siguiente:

2.10.1. Ejemplo de RIGHT JOIN

```
select est.nombre, col.nombre_colegio  
from Estudiante as est  
right join Colegio as col on est.id_colegio = col.id_colegio;
```

2.10.2. Adjuntar una imagen de conjuntos y la consulta SQL que refleje el RIGHT JOIN

	nombre	nombre_colegio
1	Carlos	31 de octubre
2	Ana	31 de octubre
3	Pedro	illimani 26 de abril
4	Sandra	san antonio
5	Saul	jardines del sur
6	<null>	italia

3. Manejo de consultas

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

```
select jug.*  
from Jugador as jug  
where jug.id_equipo = 'equ-222';
```

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

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

```
select jug.*  
from Jugador as jug  
where jug.id_equipo = 'equ-333';
```

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

3.3. Mostrar aquellos jugadores mayores o igual a 21 años

```
select jug.*  
from Jugador as jug  
where jug.edad >= '21';
```

	id_jugador	nombres	apellidos	ci	edad	id_equipo
1	jug-333	Saul	Araj	8997833LP	21	equ-222
2	jug-555	Ana	Mica	8997855LP	23	equ-333

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

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

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

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

```
select eq.*  
from Equipo as eq  
where eq.id_campeonato = 'camp-111' and eq.categoria = 'MUJERES';
```

	id_equipo	nombre_equipo	categoria	id_campeonato
1	equ-333	girls unifranz	MUJERES	camp-111

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

```
select eq.nombre_equipo  
from Equipo as eq  
inner join Jugador as jug on eq.id_equipo = jug.id_equipo  
where jug.id_jugador = 'jug-333';
```

	nombre_equipo
1	404 not found

3.7. Mostrar el nombre del campeonato del jugador con id_jugador igual a jug-333

```
select cam.nombre_campeonato  
from Campeonato as cam  
inner join Equipo as eq on cam.id_campeonato = eq.id_campeonato  
inner join Jugador as jug on jug.id_equipo = eq.id_equipo  
where jug.id_jugador = 'jug-333';
```

	nombre_campeonato
1	Campeonato Unifranz

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

```
select*  
from Campeonato as cam  
    inner join Equipo as eq on cam.id_campeonato = eq.id_campeonato  
    inner join Jugador as jug on jug.id_equipo = eq.id_equipo;
```

	cam.id_campeonato	nombre_campeonato	sede	eq.id_equipo	nombre_equipo	categoria	eq.id_campeonato	id_jugador	nombres	apellidos	ci	edad	jug.id_equipo
1	camp-111	Campeonato Unifranz	El Alto	equ-222	404 not found	VARONES	camp-111	jug-111	Carlos	Villa	8997811LP	19	equ-222
2	camp-111	Campeonato Unifranz	El Alto	equ-222	404 not found	VARONES	camp-111	jug-222	Pedro	Salas	8997822LP	20	equ-222
3	camp-111	Campeonato Unifranz	El Alto	equ-222	404 not found	VARONES	camp-111	jug-333	Saul	Araj	8997833LP	21	equ-222
4	camp-111	Campeonato Unifranz	El Alto	equ-333	girls unifranz	MUJERES	camp-111	jug-444	Sandra	Solis	8997844LP	20	equ-333
5	camp-111	Campeonato Unifranz	El Alto	equ-333	girls unifranz	MUJERES	camp-111	jug-555	Ana	Mica	8997855LP	23	equ-333

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

3.9.1. Podría utilizar la función de agregación COUNT

```
SELECT COUNT(*) as equipos_inscritos  
FROM Equipo as eq;
```

	equipos_inscritos
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

```
select COUNT(*) as CANTIDAD_VARONES  
from Equipo as eq  
where eq.categoria = 'VARONES';
```

CANTIDAD_VARONES	
1	2

```
select COUNT(*) as CANTIDAD_MUJERES  
from Equipo as eq  
where eq.categoria = 'MUJERES';
```

CANTIDAD_MUJERES	
1	1



FIN