

HITO 3

# PROCESUAL HITO 3

# BASE DE DATOS I - 2022

Estudiante: Jhon Deyvid Ajoruro Condori  
Docente: William Roddy Barra Paredes  
Materia: Base de datos 1  
Semestre: 2do

**CREAR UN  
CAMPEONATO EN  
DONDE PUEDAN  
PARTICIPAR TODAS  
LAS SEDES, EN EL  
CAMPEONATO  
PUEDEN  
INSCRIBIRSE  
TANTO CATEGORÍA  
VARONES Y  
MUJERES.**

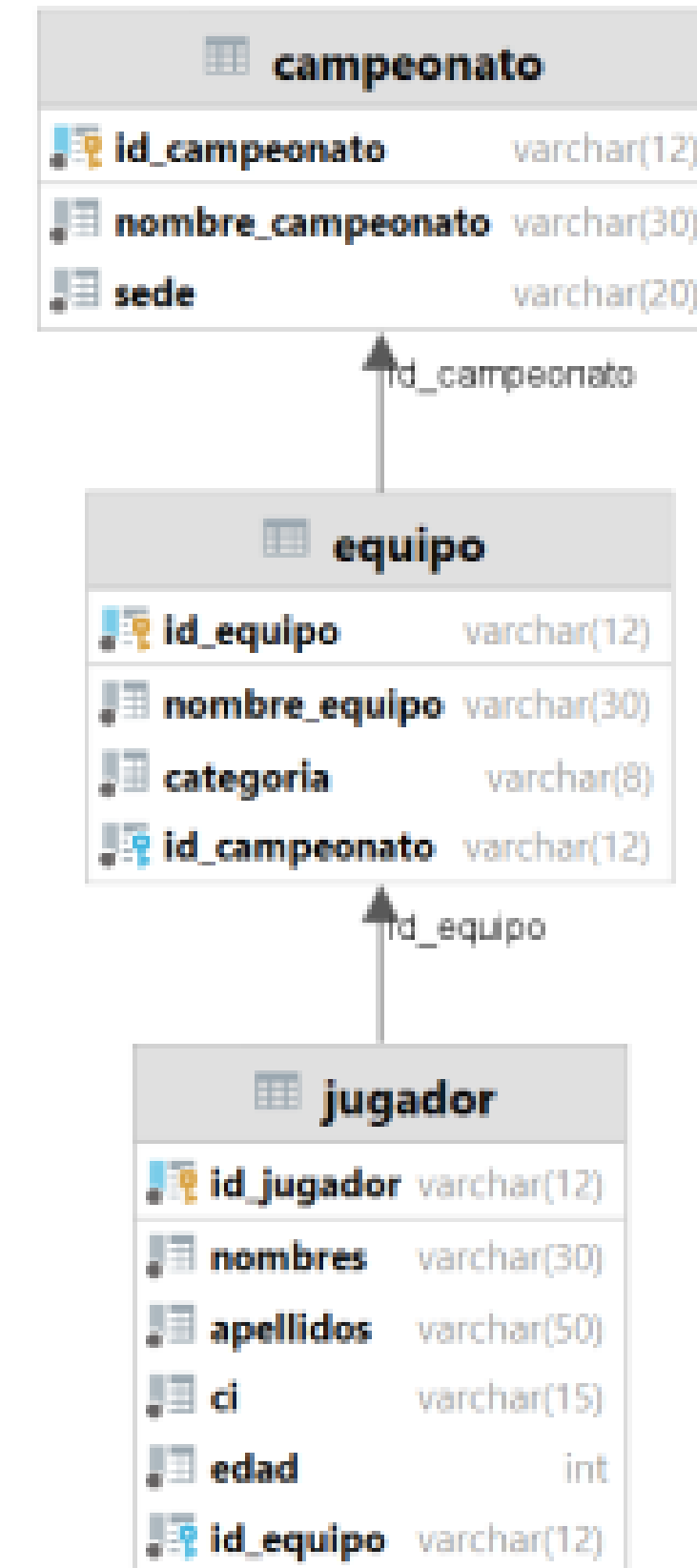


tabla campeonato		
id_campeonato	nombre_campeonato	sede
camp-111	Campeonato Unifranz	El Alto
camp-222	Campeonato Unifranz	Cochabamba

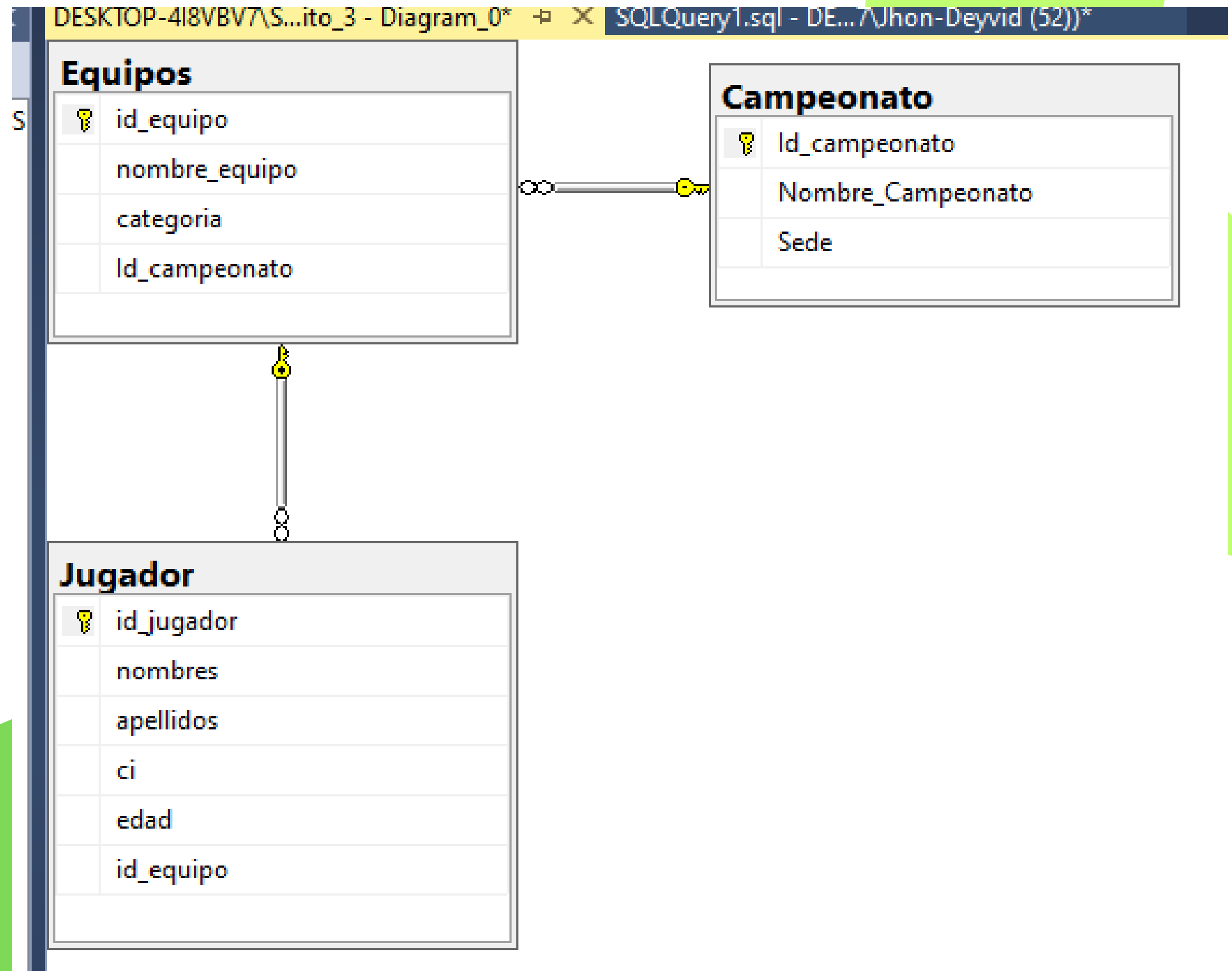
id_equipo	nombre_equipo	categoria	id_campeonato
equ-111	Google	VARONES	camp-111
equ-222	404 Not found	VARONES	camp-111
equ-333	girls unifranz	MUJERES	camp-111

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

1.2. LOS  
REGISTROS  
DE CADA  
TABLA  
DEBERÍAN  
QUEDAR DE  
LA SIGUIENTE  
FORMA

## 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 DML, ADICIONALMENTE.

- **DDL (Data Definition Language):** Se utiliza para definir estructuras de base de datos, como crear, modificar o eliminar tablas.
- **DML (Data Manipulation Language):** Se utiliza para manipular datos en las tablas, como insertar, actualizar o eliminar registros.

## 2.3. QUE SIGNIFICA PRIMARY KEY Y FOREIGN KEY

- **PRIMARY KEY:** Es una restricción que identifica de forma única cada registro en una tabla de base de datos. Cada tabla puede tener solo una clave primaria.
- **FOREIGN KEY:** Es una restricción que se utiliza para vincular dos tablas entre sí a través de una clave externa en una de las tablas. La clave externa generalmente apunta a la clave primaria de otra tabla.

## 2.4 DEFINA QUE ES UNA TABLA Y EL USO DE IDENTITY. ?

- TABLA: En una base de datos relacional, una tabla es una estructura que organiza los datos en filas y columnas. Cada columna tiene un nombre único y un tipo de datos específico.
- IDENTITY: En SQL Server, `IDENTITY` se utiliza para generar valores de clave primaria únicos automáticamente.

## 2.5 PARA QUE SE UTILIZA LA CLÁUSULA WHERE.

La cláusula `WHERE` se utiliza en las instrucciones SQL para filtrar las filas que cumplen con una condición específica.

## 2.6 PARA QUE SE UTILIZA LA INSTRUCCIÓN INNER JOIN.

La instrucción `INNER JOIN` se utiliza para mostrar filas de dos o más tablas basándose en una condición de relación entre ellas.





## **OTROS PUNTOS**

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

**2.7.1. Ejemplo de INNER JOIN**

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

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

**2.8.1. Ejemplo de LEFT JOIN**

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

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

**2.9.1. Ejemplo de RIGHT JOIN**

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

**2.10. Crear 3 tablas y crear una consulta SQL que muestra el uso de INNER JOIN.**



# COMANDO PARA INTRODUCIR DATOS

-- Insertar registros en la tabla - -

```
INSERT INTO Caracteristicas (Cedula_de_Identidad,  
Nombre_y_apellido, Edad)
```

```
VALUES (1245865, 'Juan Garcia Roque', 30),
```

```
INSERT INTO Caracteristicas (Cedula_de_Identidad,  
Nombre_y_apellido, Edad)
```

```
VALUES (2783295, 'María Galindo Flores', 25), INSERT INTO
```

**¿Como se elimina una tabla?**

```
DROP TABLE "NombreDeTabla";
```



### 3. MANEJO DE CONSULTAS

3.1. Mostrar que jugadores que son del equipo equ-222

```
SELECT nombres, apellidos  
FROM Jugador  
WHERE id_equipo = 'equ-222';
```

3.2. Mostrar que jugadores(nombres, apellidos) que juegan en la sede de El Alto.

```
SELECT nombres, apellidos  
FROM Jugador  
INNER JOIN Equipos ON Jugador.id_equipo = Equipos.id_equipo  
INNER JOIN Campeonato ON Equipos.Id_campeonato = Campeonato.Id_campeonato  
WHERE Campeonato.Sede = 'El Alto';
```

3.3. Mostrar aquellos jugadores mayores o igual a 21 años que sean de la categoría VARONES.

```
SELECT nombres, apellidos  
FROM Jugador  
WHERE edad >= 21 AND id_equipo IN (SELECT id_equipo FROM Equipos WHERE categoria = 'varones');
```

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

**3.4.1. Podría utilizar la instrucción LIKE**

```
SELECT nombres, apellidos  
FROM Jugador  
WHERE apellidos LIKE 'S%';
```

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

```
SELECT nombre_equipo  
FROM Equipos  
WHERE Id_campeonato = 'camp-111' AND categoria = 'mujeres';
```

**3.6. Mostrar el nombre del equipo del jugador con id\_jugador igual a jug-333**

```
SELECT Equipos.nombre_equipo  
FROM Jugador  
INNER JOIN Equipos ON Jugador.id_equipo = Equipos.id_equipo  
WHERE id_jugador = 'jug-333';
```

### 3.7. Mostrar el nombre del campeonato del jugador con id\_jugador igual a jug-333

```
SELECT Campeonato.Nombre_Campeonato  
FROM Jugador  
INNER JOIN Equipos ON Jugador.id_equipo = Equipos.id_equipo  
INNER JOIN Campeonato ON Equipos.Id_campeonato = Campeonato.Id_campeonato  
WHERE id_jugador = 'jug-333';
```

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

```
SELECT Jugador.nombres AS NombreJugador, Jugador.apellidos AS ApellidoJugador,  
Equipos.nombre_equipo AS NombreEquipo, Campeonato.Nombre_Campeonato AS  
NombreCampeonato  
FROM Jugador  
INNER JOIN Equipos ON Jugador.id_equipo = Equipos.id_equipo  
INNER JOIN Campeonato ON Equipos.Id_campeonato = Campeonato.Id_campeonato;
```

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

**Utilizaría el comando 'count' y el 'as' para crear una columna que se llame "cantidad de equipos"**

**3.9.1. Podría utilizar la función de agregación COUNT**

**SELECT COUNT(\*) AS CantidadEquiposInscritos**

**FROM Equipos;**

**3.10. ¿Qué estrategia utilizaría para determinar cuántos jugadores pertenecen a la categoría VARONES o Categoría MUJERES.**

**Usando un count para contar y un where para que diferencie entre hombres y mujeres.**

**3.10.1. Para esto puede utilizar la función de agregación COUNT**

**-- Contar jugadores de la categoría VARONES**

**SELECT COUNT(\*) AS TotalVarones**

**FROM Jugador**

**WHERE id\_equipo IN (SELECT id\_equipo FROM Equipos WHERE categoria = 'varones');**

**-- Contar jugadores de la categoría MUJERES**

**SELECT COUNT(\*) AS TotalMujeres**

**FROM Jugador**

**WHERE id\_equipo IN (SELECT id\_equipo FROM Equipos WHERE categoria = 'mujeres');**



**PRESENTACION  
CONCLUIDA  
GRACIAS POR SU  
ATENCION**