



# BASE DE DATOS – SQL SERVER

2023



Internacionalízate

**INNOVACIÓN  
EN EDUCACIÓN**



# CONTENIDO

Qué *incluye* este *informe*

- 
- 01      Diseño de base de datos

---

  - 02      Manejo de conceptos

---

  - 03      Manejo de consultas

---

ESTUDIANTE : KEVIN BRAYAN MENDOZA CHEJO

INGENIERIA DE SISTEMAS

# 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.



# PASO 1

```
-- Crear la tabla campeonato  
CREATE TABLE campeonato (  
    id_campeonato varchar(12),  
    campeonato varchar(30),  
    sede varchar(12)  
);
```

```
-- Crear la tabla equipo
```

```
CREATE TABLE equipo (  
    id_equipo varchar(12),  
    nombre_equipo varchar(30),  
    id_campeonato varchar(12)  
);
```

```
-- Crear la tabla jugador
```

```
CREATE TABLE jugador (  
    id_jugador varchar(12),  
    nombres varchar(50),  
    apellidos varchar(15),  
    edad varchar(30),  
    id_equipo varchar(12)  
);
```

# PASO 2

```
-- Insertar datos en la tabla campeonato  
INSERT INTO campeonato (id_campeonato, campeonato,  
                        sede)  
VALUES  
      ('1', 'Liga A', 'LE1'),  
      ('2', 'Liga B', 'LE2'),  
      ('3', 'Liga C', 'LE3'),  
      ('4', 'Liga D', 'LE4'),  
      ('5', 'Liga E', 'LE5');
```

```
-- Insertar datos en la tabla equipo
```

```
INSERT INTO equipo (id_equipo, nombre_equipo,  
                   id_campeonato)  
VALUES  
      ('E1', 'Equipo 1', '1'),  
      ('E2', 'Equipo 2', '1'),  
      ('E3', 'Equipo 3', '2'),  
      ('E4', 'Equipo 4', '2'),  
      ('E5', 'Equipo 5', '3');
```

```
-- Insertar datos en la tabla jugador
```

```
INSERT INTO jugador (id_jugador, nombres, apellidos,  
                     edad, id_equipo)  
VALUES  
      ('J1', 'Juan', 'Perez', '25', 'E1'),  
      ('J2', 'Maria', 'Gomez', '28', 'E1'),  
      ('J3', 'Luis', 'Lopez', '22', 'E2'),  
      ('J4', 'Ana', 'Martinez', '26', 'E3'),  
      ('J5', 'Carlos', 'Rodriguez', '24', 'E4');
```

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



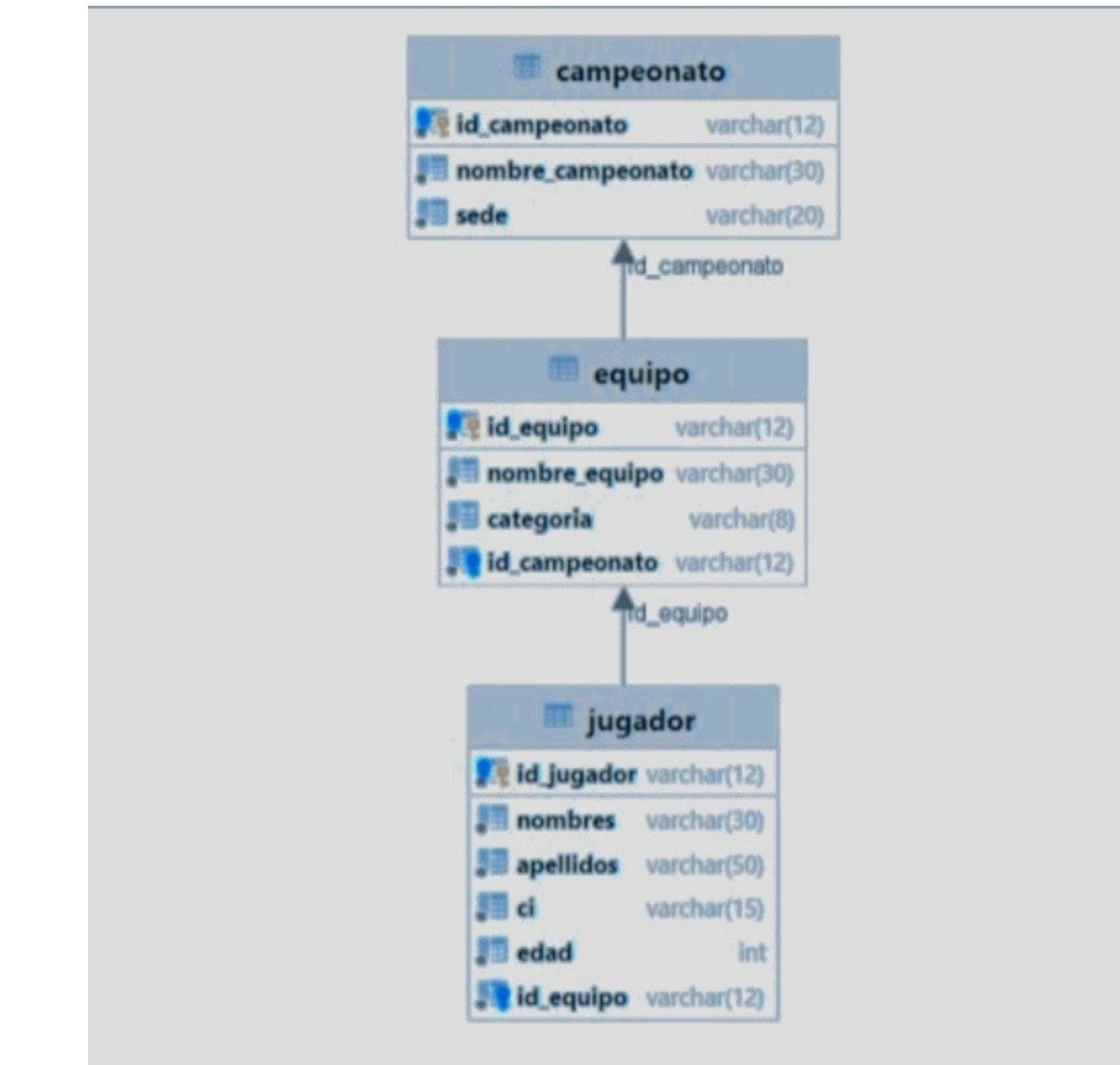
-- Mostrar los datos de la tabla campeonato  
SELECT \* FROM campeonato;

-- Mostrar los datos de la tabla equipo  
SELECT \* FROM equipo;

-- Mostrar los datos de la tabla jugador  
SELECT \* FROM jugador;

## 2. Manejo de conceptos

### 2.1. Adjuntar el diagrama E-R GENERADO por su editor (DATAGRIP o SQL SERVER MANAGEMENTS STUDIO)



## 2. Manejo de conceptos

### 2.2. Que es DDL y DML, adicionalmente muestra un ejemplo en la base de datos UNIFRANZITOS.

DDL (Data Definition Language): Estos comandos se utilizan para definir y gestionar la estructura de la base de datos, como crear, modificar o eliminar tablas, índices y restricciones.

```
-- Comando DDL para crear una tabla  
CREATE TABLE Estudiantes (  
    EstudianteID int PRIMARY KEY,  
    Nombre varchar(50),  
    Edad int,  
    Carrera varchar(50)  
);
```

DML (Data Manipulation Language): Estos comandos se utilizan para manipular los datos dentro de la base de datos, como insertar, actualizar, eliminar y recuperar registros.

```
-- Comando DML para insertar un nuevo estudiante  
INSERT INTO Estudiantes (EstudianteID,  
    Nombre, Edad, Carrera)  
VALUES (1, 'Juan Perez', 20,  
    'Ingeniería Informática');
```



## 2. Manejo de conceptos

### 2.3. Que significa PRIMARY KEY y FOREIGN KEY

#### PRIMARY KEY

PRIMARY KEY (Clave Primaria): Una PRIMARY KEY es un atributo o conjunto de atributos en una tabla que identifica de manera única cada fila o registro en esa tabla.

#### FOREIGN KEY

FOREIGN KEY (Clave Foránea): Una FOREIGN KEY es un atributo en una tabla que se relaciona con la PRIMARY KEY de otra tabla.



## 2. Manejo de conceptos

### 2.4. Defina que es una TABLA y el uso de IDENTITY.

Una tabla en una base de datos es una estructura que organiza datos en filas y columnas. La propiedad "IDENTITY" se usa para generar automáticamente valores únicos en una columna, como identificadores, sin necesidad de ingresarlos manualmente.

### 2.5. Para que se utiliza la cláusula WHERE.

La cláusula "WHERE" se utiliza para filtrar y seleccionar registros específicos de una base de datos en función de condiciones específicas, lo que permite obtener datos relevantes de acuerdo con ciertos criterios.

### 2.6. Para que se utiliza la instrucción INNER JOIN.

La instrucción "INNER JOIN" se utiliza para combinar datos de dos o más tablas en una base de datos, permitiendo la recuperación de información relacionada y completa en una sola consulta.

## 2. Manejo de conceptos



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

EJEMPLOS.

#### 2.7.1. Ejemplo de INNER JOIN

```
SELECT Clientes.Nombre,  
Clientes.Apellido, Pedidos.Producto,  
Pedidos.Cantidad  
FROM Clientes  
INNER JOIN Pedidos ON  
Clientes.ClienteID = Pedidos.ClienteID;
```

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

```
SELECT Clientes.Nombre,  
Clientes.Apellido, Pedidos.Producto,  
Pedidos.Cantidad  
FROM Clientes  
INNER JOIN Pedidos ON  
Clientes.ClienteID =  
Pedidos.ClienteID;
```



## 2. Manejo de conceptos

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

### **2.8.1. Ejemplo de LEFT JOIN**

```
SELECT Clientes.Nombre,  
Clientes.Apellido, Pedidos.Producto,  
Pedidos.Cantidad  
FROM Clientes  
LEFT JOIN Pedidos ON  
Clientes.ClienteID =  
Pedidos.ClienteID;
```

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

```
SELECT Clientes.Nombre,  
Clientes.Apellido, Pedidos.Producto,  
Pedidos.Cantidad  
FROM Clientes  
LEFT JOIN Pedidos ON  
Clientes.ClienteID =  
Pedidos.ClienteID;
```



## 2. Manejo de conceptos

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

### 2.9.1. Ejemplo de RIGHT JOIN

```
SELECT Clientes.Nombre,  
Clientes.Apellido, Pedidos.Producto,  
Pedidos.Cantidad  
FROM Clientes  
RIGHT JOIN Pedidos ON  
Clientes.ClienteID =  
Pedidos.ClienteID;
```

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

```
SELECT Clientes.Nombre,  
Clientes.Apellido, Pedidos.Producto,  
Pedidos.Cantidad  
FROM Clientes  
RIGHT JOIN Pedidos ON  
Clientes.ClienteID =  
Pedidos.ClienteID;
```

## 2. Manejo de conceptos

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

```
CREATE TABLE Clientes (
    ClienteID INT PRIMARY KEY,
    Nombre VARCHAR(50),
    Apellido VARCHAR(50)
);
```

```
CREATE TABLE Pedidos (
    PedidoID INT PRIMARY KEY,
    ClienteID INT,
    FechaPedido DATE,
    Total DECIMAL(10, 2)
);
```

```
CREATE TABLE Productos (
    ProductoID INT PRIMARY KEY,
    NombreProducto VARCHAR(100),
    Precio DECIMAL(8, 2)
);
```

-- Consulta con INNER JOIN para combinar datos

```
SELECT Clientes.Nombre, Clientes.Apellido, Pedidos.FechaPedido, Productos.NombreProducto, Productos.Precio
FROM Clientes
INNER JOIN Pedidos ON Clientes.ClienteID = Pedidos.ClienteID
INNER JOIN Productos ON Pedidos.PedidoID = Productos.ProductoID;
```

# 3. Manejo de consultas



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

```
SELECT Jugador.Rid_jugador  
FROM Jugador  
INNER JOIN Campeonato ON  
Jugador.Rid_jugador =  
Campeonato.Rid_jugador  
WHERE Campeonato.Rid_equipo =  
'equ-222';
```

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

```
SELECT Campeonato.nombre_equipo,  
Jugador.Rid_jugador  
FROM Campeonato  
INNER JOIN Jugador ON  
Campeonato.Rid_jugador =  
Jugador.Rid_jugador  
WHERE Campeonato.sede = 'El Alto';
```

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

```
SELECT Jugador.Rid_jugador  
FROM Jugador  
INNER JOIN Campeonato ON  
Jugador.Rid_jugador =  
Campeonato.Rid_jugador  
WHERE Campeonato.categoría =  
'VARONES' AND DATEDIFF(YEAR,  
Campeonato.ci, GETDATE()) >= 21;
```

# 3. Manejo de consultas



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

```
SELECT nombre_equipo, apellidos  
FROM Campeonato  
WHERE apellidos LIKE 'S%';
```

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

```
SELECT DISTINCT nombre_equipo  
FROM Campeonato  
WHERE Rid_campeonato = 'camp-111' AND  
categoria = 'MUJERES';
```

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

```
SELECT Campeonato.nombre_equipo  
FROM Campeonato  
INNER JOIN Jugador ON  
Campeonato.Rid_jugador =  
Jugador.Rid_jugador  
WHERE Jugador.Rid_jugador =  
'jug-333';
```

### 3. Manejo de consultas



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

```
SELECT Campeonato.nombre_campeonato  
FROM Campeonato  
INNER JOIN Jugador ON  
Campeonato.Rid_jugador =  
Jugador.Rid_jugador  
WHERE Jugador.Rid_jugador =  
'jug-333';
```

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

```
SELECT Jugador.Rid_jugador,  
Campeonato.Rid_campeonato,  
Campeonato.nombre_equipo  
FROM Jugador  
INNER JOIN Campeonato ON  
Jugador.Rid_jugador =  
Campeonato.Rid_jugador;
```



### 3. Manejo de consultas

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

```
SELECT COUNT(DISTINCT nombre_equipo)  
AS TotalEquiposInscritos  
FROM Campeonato;
```

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

```
SELECT categoria, COUNT(*) AS  
TotalJugadoresPorCategoria  
FROM Campeonato  
GROUP BY categoria;
```

# ¡Gracias!

No dudes en ponerte en contacto con nosotros si tienes alguna duda.

**UNIFRANZ**

Internacionalízate

**INNOVACIÓN  
EN EDUCACIÓN**

