

```

-- Crear la tabla de Torneos
CREATE TABLE Torneos (
    TorneoID INTEGER PRIMARY KEY,
    Nombre TEXT,
    Ubicacion TEXT,
    FechaInicio DATE,
    FechaFin DATE
);

-- Crear la tabla de Participantes
CREATE TABLE Participantes (
    ParticipanteID INTEGER PRIMARY KEY,
    Nombre TEXT,
    Edad INTEGER,
    Nacionalidad TEXT
);

-- Insertar datos en la tabla de Participantes
INSERT INTO Participantes (ParticipanteID, Nombre, Edad, Nacionalidad) VALUES
(1, 'Juan Pérez', 28, 'España'),
(2, 'María Gómez', 24, 'Argentina'),
(3, 'Carlos Ruiz', 30, 'México'),
(4, 'Ana López', 26, 'Chile'),
(5, 'Pedro Martínez', 22, 'Perú'),
(6, 'Laura Sánchez', 29, 'Colombia'),
(7, 'Luis Fernández', 25, 'Brasil'),
(8, 'Marta Díaz', 27, 'Venezuela');

-- Insertar datos en la tabla de Torneos
INSERT INTO Torneos (TorneoID, Nombre, Ubicacion, FechaInicio, FechaFin) VALUES
(1, 'Open de España', 'Madrid', '2024-06-01', '2024-06-10'),
(2, 'Copa América', 'Buenos Aires', '2024-07-01', '2024-07-08');

-- Crear la tabla de Partidos con claves foráneas
CREATE TABLE Partidos (
    PartidoID INTEGER PRIMARY KEY,
    TorneoID INTEGER,
    Participante1ID INTEGER,
    Participante2ID INTEGER,
    Fecha DATE,
    FOREIGN KEY (TorneoID) REFERENCES Torneos(TorneoID),
    FOREIGN KEY (Participante1ID) REFERENCES Participantes(ParticipanteID),
    FOREIGN KEY (Participante2ID) REFERENCES Participantes(ParticipanteID)
);

-- Insertar datos en la tabla de Partidos
INSERT INTO Partidos (PartidoID, TorneoID, Participante1ID, Participante2ID, Fecha) VALUES
(1, 1, 1, 2, '2024-06-01'),
(2, 1, 3, 4, '2024-06-02'),
(3, 2, 5, 6, '2024-07-01'),
(4, 2, 7, 8, '2024-07-02');

```

## CONSULTAS

### INNER JOIN

El INNER JOIN devuelve las filas cuando hay una coincidencia en ambas tablas.

Ejemplo 1: Obtener los detalles de todos los partidos junto con la información del torneo correspondiente:

```
SELECT Partidos.PartidoID, Torneos.Nombre AS TorneoNombre, Torneos.Ubicacion,
Partidos.Fecha
FROM Partidos
INNER JOIN Torneos ON Partidos.TorneoID = Torneos.TorneoID;
```

Ejemplo 2: Obtener los detalles de los partidos junto con la información de los participantes:

```
SELECT
    Partidos.PartidoID,
    Torneos.Nombre AS TorneoNombre,
    Participante1.Nombre AS Participante1Nombre,
    Participante2.Nombre AS Participante2Nombre,
    Partidos.Fecha
FROM Partidos
INNER JOIN Torneos ON Partidos.TorneoID = Torneos.TorneoID
INNER JOIN Participantes AS Participante1 ON Partidos.Participante1ID =
Participante1.ParticipanteID
INNER JOIN Participantes AS Participante2 ON Partidos.Participante2ID =
Participante2.ParticipanteID;
```

### LEFT JOIN

El LEFT JOIN devuelve todas las filas de la tabla de la izquierda (Partidos), y las filas coincidentes de la tabla de la derecha (Torneos). Si no hay coincidencia, el resultado es NULL desde la tabla de la derecha.

Ejemplo: Obtener todos los partidos y la información del torneo, incluyendo los partidos que no tienen torneo asociado:

```
SELECT Partidos.PartidoID, Partidos.TorneoID, Torneos.Nombre AS TorneoNombre,
Partidos.Fecha
FROM Partidos
LEFT JOIN Torneos ON Partidos.TorneoID = Torneos.TorneoID;
```

### Emulación de RIGHT JOIN

Ejemplo de Emulación de RIGHT JOIN: Obtener todos los torneos y la información de los partidos, incluyendo los torneos que no tienen partidos asociados:

Para este ejemplo, vamos a suponer que hay un torneo que no tiene ningún partido asociado en la tabla Partidos.

\*\*\*

```
INSERT INTO Torneos (TorneoID, Nombre, Ubicacion, FechaInicio, FechaFin) VALUES
```

```
(3, 'Torneo Fantasma', 'Desconocida', '2024-08-01', '2024-08-05');
```

\*\*\*

```
-- Emulación de RIGHT JOIN
```

```
SELECT Partidos.PartidoID, Torneos.TorneoID, Torneos.Nombre AS TorneoNombre,  
Partidos.Fecha  
FROM Torneos  
LEFT JOIN Partidos ON Torneos.TorneoID = Partidos.TorneoID;
```

En este caso, el resultado incluirá el "Torneo Fantasma" aunque no tenga partidos asociados, mostrando NULL para los campos de la tabla Partidos.

Emulación de FULL OUTER JOIN

Ejemplo de Emulación de FULL OUTER JOIN: Obtener todos los torneos y todos los partidos, combinando ambas tablas y mostrando NULLs donde no haya coincidencias:

Para este ejemplo, vamos a suponer que además de un torneo sin partidos, hay un partido que no tiene torneo asociado (esto puede suceder si, por error, un partido se insertó sin una referencia válida a un torneo).

\*\*\*

```
INSERT INTO Partidos (PartidoID, TorneoID, Participante1ID, Participante2ID, Fecha) VALUES  
(5, NULL, 1, 3, '2024-08-01');
```

\*\*\*

```
-- Emulación de FULL OUTER JOIN
```

```
SELECT Partidos.PartidoID, Torneos.TorneoID, Torneos.Nombre AS TorneoNombre,  
Partidos.Fecha  
FROM Partidos  
LEFT JOIN Torneos ON Partidos.TorneoID = Torneos.TorneoID
```

UNION

```
SELECT Partidos.PartidoID, Torneos.TorneoID, Torneos.Nombre AS TorneoNombre,  
Partidos.Fecha  
FROM Torneos  
LEFT JOIN Partidos ON Torneos.TorneoID = Partidos.TorneoID;
```

cuarta tabla

```
-- Crear la tabla de sponsors
```

```
CREATE TABLE Sponsors (  
    SponsorID INTEGER PRIMARY KEY,  
    Nombre TEXT
```

);

-- Insertar datos de sponsors

INSERT INTO Sponsors (SponsorID, Nombre) VALUES

(1, 'Nike'),  
(2, 'Adidas'),  
(3, 'Puma'),  
(4, 'Reebok'),  
(5, 'Under Armour'),  
(6, 'New Balance'),  
(7, 'Asics'),  
(8, 'Fila'),  
(9, 'Lotto'),  
(10, 'Kappa');

-- Crear tabla de relación entre jugadores y sponsors

CREATE TABLE Jugador\_Sponsor (  
    JugadorID INTEGER,  
    SponsorID INTEGER,  
    PRIMARY KEY (JugadorID, SponsorID),  
    FOREIGN KEY (JugadorID) REFERENCES Participantes(ParticipanteID),  
    FOREIGN KEY (SponsorID) REFERENCES Sponsors(SponsorID)  
);

-- Insertar datos de relación entre jugadores y sponsors

INSERT INTO Jugador\_Sponsor (JugadorID, SponsorID) VALUES

(1, 1), (1, 2), (1, 3); -- Juan Pérez patrocinado por Nike, Adidas y Puma

INSERT INTO Jugador\_Sponsor (JugadorID, SponsorID) VALUES

(2, 2), (2, 3), (2, 4); -- María Gómez patrocinada por Adidas, Puma y Reebok

INSERT INTO Jugador\_Sponsor (JugadorID, SponsorID) VALUES

(3, 3), (3, 4), (3, 5); -- Carlos Ruiz patrocinado por Puma, Reebok y Under Armour

INSERT INTO Jugador\_Sponsor (JugadorID, SponsorID) VALUES

(4, 4), (4, 5), (4, 6); -- Ana López patrocinada por Reebok, Under Armour y New Balance

INSERT INTO Jugador\_Sponsor (JugadorID, SponsorID) VALUES

(5, 5), (5, 6), (5, 7); -- Pedro Martínez patrocinado por Under Armour, New Balance y Asics

INSERT INTO Jugador\_Sponsor (JugadorID, SponsorID) VALUES

(6, 6), (6, 7), (6, 8); -- Laura Sánchez patrocinada por New Balance, Asics y Fila

INSERT INTO Jugador\_Sponsor (JugadorID, SponsorID) VALUES

(7, 7), (7, 8), (7, 9); -- Luis Fernández patrocinado por Asics, Fila y Lotto

INSERT INTO Jugador\_Sponsor (JugadorID, SponsorID) VALUES

(8, 8), (8, 9), (8, 10); -- Marta Díaz patrocinada por Fila, Lotto y Kappa

```
INSERT INTO Jugador_Sponsor (JugadorID, SponsorID) VALUES  
(9, 9), (9, 10), (9, 1); -- David Torres patrocinado por Lotto, Kappa y Nike
```

```
INSERT INTO Jugador_Sponsor (JugadorID, SponsorID) VALUES  
(10, 10), (10, 1), (10, 2); -- Sofía Martínez patrocinada por Kappa, Nike y Adidas
```

## SUBCONSULTAS

Ejemplo 1: Obtener el nombre y la edad del participante más joven

```
SELECT Nombre, Edad  
FROM Participantes  
WHERE Edad = (SELECT MIN(Edad) FROM Participantes);
```

Ejemplo 2: Obtener los nombres de los participantes que han jugado en el "Open de España"

```
SELECT DISTINCT p.Nombre  
FROM Participantes p  
WHERE p.ParticipanteID IN (  
    SELECT pa.Participante1ID FROM Partidos pa  
    WHERE pa.TorneoID = (SELECT TorneoID FROM Torneos WHERE Nombre = 'Open de  
España')  
    UNION  
    SELECT pa.Participante2ID FROM Partidos pa  
    WHERE pa.TorneoID = (SELECT TorneoID FROM Torneos WHERE Nombre = 'Open de  
España')  
);
```

Ejemplo 3: Obtener los detalles de los partidos que tienen participantes patrocinados por "Nike"

```
SELECT p.PartidoID, t.Nombre AS TorneoNombre, part1.Nombre AS Participante1Nombre,  
part2.Nombre AS Participante2Nombre, p.Fecha  
FROM Partidos p  
INNER JOIN Torneos t ON p.TorneoID = t.TorneoID  
INNER JOIN Participantes part1 ON p.Participante1ID = part1.ParticipanteID  
INNER JOIN Participantes part2 ON p.Participante2ID = part2.ParticipanteID  
WHERE part1.ParticipanteID IN (SELECT JugadorID FROM Jugador_Sponsor WHERE  
SponsorID = (SELECT SponsorID FROM Sponsors WHERE Nombre = 'Nike'))  
OR part2.ParticipanteID IN (SELECT JugadorID FROM Jugador_Sponsor WHERE SponsorID =  
(SELECT SponsorID FROM Sponsors WHERE Nombre = 'Nike'));
```

Ejemplo 4: Listar los torneos que no tienen partidos asociados

```
SELECT Nombre, Ubicacion, FechaInicio, FechaFin  
FROM Torneos  
WHERE TorneoID NOT IN (SELECT DISTINCT TorneoID FROM Partidos);
```

Ejemplo 5: Encontrar los nombres de los sponsors que patrocinan a más de tres jugadores

```
SELECT s.Nombre  
FROM Sponsors s
```

```
WHERE (SELECT COUNT(*) FROM Jugador_Sponsor js WHERE js.SponsorID = s.SponsorID)
> 3;
```

Ejemplo 6: Obtener el número de partidos que ha jugado cada participante

```
SELECT p.Nombre, (
    SELECT COUNT(*)
    FROM Partidos pa
    WHERE pa.Participante1ID = p.ParticipanteID OR pa.Participante2ID = p.ParticipanteID
) AS NumeroDePartidos
FROM Participantes p;
```

Ejemplo 7: Listar los torneos y la cantidad de partidos asociados a cada uno

```
SELECT t.Nombre, t.Ubicacion, (
    SELECT COUNT(*)
    FROM Partidos pa
    WHERE pa.TorneoID = t.TorneoID
) AS NumeroDePartidos
FROM Torneos t;
```

Ejemplo 8: Obtener los nombres de los participantes que tienen más de un sponsor

```
SELECT p.Nombre
FROM Participantes p
WHERE (SELECT COUNT(*) FROM Jugador_Sponsor js WHERE js.JugadorID =
p.ParticipanteID) > 1;
```

Ejemplo 9: Encontrar los nombres de los torneos donde participó "Juan Pérez"

```
SELECT DISTINCT t.Nombre
FROM Torneos t
WHERE t.TorneoID IN (
    SELECT pa.TorneoID FROM Partidos pa
    WHERE pa.Participante1ID = (SELECT ParticipanteID FROM Participantes WHERE Nombre =
'Juan Pérez')
    OR pa.Participante2ID = (SELECT ParticipanteID FROM Participantes WHERE Nombre =
'Juan Pérez')
);
```

Ejemplo 10: Listar los nombres de los torneos que tuvieron al menos un partido el '2024-06-01'

```
SELECT DISTINCT t.Nombre
FROM Torneos t
WHERE t.TorneoID IN (
    SELECT p.TorneoID
    FROM Partidos p
    WHERE p.Fecha = '2024-06-01'
);
```

## AÑADIR GENERO A LAS TABLAS

Añadir la columna Genero a la tabla Participantes

```
ALTER TABLE Participantes ADD COLUMN Genero TEXT;
```

Actualizar la tabla Participantes con los géneros (esto es solo un ejemplo y debes actualizar con los datos correctos)

```
UPDATE Participantes SET Genero = 'Hombre' WHERE ParticipanteID IN (1, 3, 5, 7, 9, 11, 13, 15, 17, 19); -- IDs de los hombres
```

```
UPDATE Participantes SET Genero = 'Mujer' WHERE ParticipanteID IN (2, 4, 6, 8, 10, 12, 14, 16, 18, 20); -- IDs de las mujeres
```

Calcular la edad promedio de los participantes

```
SELECT AVG(Edad) AS EdadPromedio FROM Participantes;
```

Calcular el porcentaje de hombres y mujeres por encima de la edad promedio

```
WITH EdadPromedio AS (  
    SELECT AVG(Edad) AS Promedio FROM Participantes  
)  
,  
ParticipantesPorEncimaPromedio AS (  
    SELECT Genero, COUNT(*) AS Total  
    FROM Participantes  
    WHERE Edad > (SELECT Promedio FROM EdadPromedio)  
    GROUP BY Genero  
)  
,  
TotalParticipantes AS (  
    SELECT Genero, COUNT(*) AS Total  
    FROM Participantes  
    GROUP BY Genero  
)  
SELECT  
    p.Genero,  
    ROUND((p.Total * 100.0 / t.Total), 2) AS Porcentaje  
FROM ParticipantesPorEncimaPromedio p  
JOIN TotalParticipantes t ON p.Genero = t.Genero;
```