

Normalización: Incluye el proceso de normalización mediante el cual el diagrama fue depurado a las 3 formas normales

Entidades:

Piloto:

piloto_id (ID único del piloto)

nombre (Nombre completo del piloto)

nacionalidad (País de origen del piloto)

fecha_nacimiento (Fecha de nacimiento del piloto)

equipo_id (ID del equipo al que pertenece el piloto)

Equipo:

equipo_id (ID único del equipo)

nombre (Nombre del equipo)

jefe_equipo (Nombre del jefe del equipo)

nacionalidad (País de origen del equipo)

Carrera:

carrera_id (ID único de la carrera)

nombre (Nombre oficial de la carrera)

circuito (Nombre del circuito donde se disputa la carrera)

fecha (Fecha de celebración de la carrera)

país (País donde se encuentra el circuito)

Resultado:

resultado_id (ID único del resultado)

carrera_id (ID de la carrera a la que pertenece el resultado)

piloto_id (ID del piloto que participa en el resultado)

posicion (Posición final del piloto en la carrera)

puntos (Puntos obtenidos por el piloto en la carrera)

Relaciones:

Un piloto pertenece a un solo equipo (relación 1:N entre Piloto y Equipo).

Una carrera tiene muchos resultados (relación 1:N entre Carrera y Resultado).

Un resultado pertenece a una sola carrera y a un solo piloto (relación N:1 entre Resultado, Carrera y Piloto).

Aplicación de la Primera Forma Normal (1FN) a la base de datos de Fórmula 1

Problema de redundancia de datos en la base de datos original:

La base de datos original presenta redundancia de datos en la tabla Piloto, específicamente en el atributo equipo_id. Esto se debe a que el atributo equipo_id almacena el nombre completo del equipo para cada piloto, en lugar de solo la referencia a la ID del equipo correspondiente.

Solución:

Para eliminar la redundancia de datos y cumplir con los principios de la 1FN, se deben realizar los siguientes cambios:

Crear una nueva tabla Equipo:

Esta tabla almacenará la información específica de cada equipo, incluyendo su nombre, jefe de equipo y nacionalidad.

SQL

CREATE TABLE Equipo (

equipo_id INT PRIMARY KEY AUTO_INCREMENT,

nombre VARCHAR(255) NOT NULL,

```
jefe_equipo VARCHAR(255),  
nacionalidad VARCHAR(255)  
);
```

Usa el código con precaución.

content_copy

Modificar la tabla Piloto:

En la tabla Piloto, se debe eliminar el atributo nombre_equipo y reemplazarlo por un atributo equipo_id que almacene la referencia a la ID del equipo correspondiente en la nueva tabla Equipo.

SQL

```
ALTER TABLE Piloto
```

```
MODIFY equipo_id INT NOT NULL;
```

-- Agregar restricción de clave foránea

```
ALTER TABLE Piloto
```

```
ADD CONSTRAINT fk_piloto_equipo
```

```
FOREIGN KEY (equipo_id) REFERENCES Equipo(equipo_id);
```

Usa el código con precaución.

content_copy

Resultado:

Con estos cambios, la base de datos se encuentra en Primera Forma Normal (1FN), eliminando la redundancia de datos y mejorando la integridad de la información.

Aplicación de la Segunda Forma Normal (2FN) a la base de datos de Fórmula 1

Problema de dependencias transitivas en la base de datos original:

La base de datos original, aunque se encuentra en Primera Forma Normal (1FN), aún presenta dependencias transitivas en la tabla Resultado. La dependencia transitiva se produce cuando un atributo no principal (en este caso, posicion y puntos) depende de otro atributo no principal (equipo_id), en lugar de depender directamente de la clave primaria (resultado_id).

Solución:

Para eliminar las dependencias transitivas y cumplir con los principios de la 2FN, se deben realizar los siguientes cambios:

Crear una nueva tabla ClasificacionCarrera:

Esta tabla almacenará la información específica de la clasificación de cada piloto en una carrera, incluyendo la posición final y los puntos obtenidos.

SQL

```
CREATE TABLE ClasificacionCarrera (  
    clasificacion_id INT PRIMARY KEY AUTO_INCREMENT,  
    resultado_id INT NOT NULL,  
    posicion INT NOT NULL,  
    puntos INT NOT NULL,  
    FOREIGN KEY (resultado_id) REFERENCES Resultado(resultado_id)  
);
```

Usa el código con precaución.

content_copy

Modificar la tabla Resultado:

En la tabla Resultado, se debe eliminar los atributos posicion y puntos y mantener solo los atributos necesarios para la relación entre la carrera y el piloto.

SQL

ALTER TABLE Resultado

DROP COLUMN posicion;

ALTER TABLE Resultado

DROP COLUMN puntos;

Usa el código con precaución.

content_copy

Resultado:

Con estos cambios, la base de datos se encuentra en Segunda Forma Normal (2FN), eliminando las dependencias transitivas y mejorando la organización de la información.

Aplicación de la Tercera Forma Normal (3FN) a la base de datos de Fórmula 1

Problema de dependencias no primarias en la base de datos original:

La base de datos original, aunque se encuentra en Segunda Forma Normal (2FN), aún presenta dependencias no primarias en la tabla ClasificacionCarrera. La dependencia no primaria se produce cuando un atributo no principal (posicion) depende de otro atributo no principal (equipo_id), en lugar de depender directamente de la clave primaria (clasificacion_id).

Solución:

Para eliminar las dependencias no primarias y cumplir con los principios de la 3FN, se deben realizar los siguientes cambios:

Crear una nueva tabla PilotoEquipoCarrera:

Esta tabla almacenará la relación entre un piloto, un equipo y una carrera específica, sin almacenar información redundante.

SQL

```
CREATE TABLE PilotoEquipoCarrera (  
    pec_id INT PRIMARY KEY AUTO_INCREMENT,  
    piloto_id INT NOT NULL,  
    equipo_id INT NOT NULL,  
    carrera_id INT NOT NULL,  
    FOREIGN KEY (piloto_id) REFERENCES Piloto(piloto_id),  
    FOREIGN KEY (equipo_id) REFERENCES Equipo(equipo_id),  
    FOREIGN KEY (carrera_id) REFERENCES Carrera(carrera_id)  
);
```

Usa el código con precaución.

content_copy

Modificar la tabla ClasificacionCarrera:

En la tabla ClasificacionCarrera, se debe eliminar el atributo equipo_id y mantener solo los atributos necesarios para la clasificación del piloto en la carrera.

SQL

```
ALTER TABLE ClasificacionCarrera  
  
DROP COLUMN equipo_id;
```

Usa el código con precaución.

content_copy

Establecer una relación entre ClasificacionCarrera y PilotoEquipoCarrera:

Se debe crear una restricción de clave foránea en la tabla ClasificacionCarrera para referenciar la tabla PilotoEquipoCarrera.

SQL

```
ALTER TABLE ClasificacionCarrera
```

```
ADD CONSTRAINT fk_clasificacion_pec
```

```
FOREIGN KEY (piloto_id, equipo_id, carrera_id) REFERENCES PilotoEquipoCarrera(piloto_id,  
equipo_id, carrera_id);
```

Usa el código con precaución.

content_copy

Resultado:

Con estos cambios, la base de datos se encuentra en Tercera Forma Normal (3FN), eliminando las dependencias no primarias y organizando la información de manera óptima.