Guía detallada de las consultas SQL para el sistema Star Trek, explicando cada operación CRUD (Create, Read, Update, Delete) con ejemplos prácticos.

Consultas de Inserción (CREATE)

1. Registro de Planetas

INSERT INTO Planets (

scientific\_name,

common\_name,

population,

galactic\_x,

galactic\_y,

galactic\_z

) VALUES (

'VUL-002', -- Nombre científico único

'Vulcano', -- Nombre común

6200000000, -- Población total

16.23, -- Coordenada X

-23.45, -- Coordenada Y

8.90 -- Coordenada Z

);

Esta inserción es fundamental porque los planetas son la base de nuestro universo Star Trek. Observemos que:

- El nombre científico sigue un formato específico (XXX-)

- Las coordenadas galácticas usan un sistema tridimensional preciso

- La población se maneja como BIGINT para soportar grandes números

2. Registro de Montañas

INSERT INTO Mountains (

planet\_id,

name,

height

) VALUES (

1, -- ID del planeta

'Monte T''Rang', -- Nombre de la montaña

12458.32 -- Altura en metros

);

Las montañas son características geográficas importantes de cada planeta. Nótese el uso de comillas dobles para manejar apóstrofes en nombres.

3. Registro de Imperios

INSERT INTO Empires (

galactic\_code,

name,

avg\_temperature

) VALUES (

'UFP-001', -- Código galáctico único

'Federación Unida de Planetas',

15.5 -- Temperatura promedio

);

Los imperios son entidades políticas fundamentales. Cada imperio tiene:

- Un código galáctico único que sigue el formato AAA-

- Una temperatura promedio que indica sus condiciones ambientales típicas

4. Registro de Flotas y Misiones

-- Primero insertamos la flota

INSERT INTO Fleets (

galactic\_code,

destination,

empire\_id

) VALUES (

'FLT-001',

'Cuadrante Alfa',

1

);

-- Luego registramos sus misiones

INSERT INTO Fleet\_Missions (

fleet\_id,

mission\_type

) VALUES

(currval('fleets\_fleet\_id\_seq'), 'Exploración'),

(currval('fleets\_fleet\_id\_seq'), 'Investigación científica');

Este es un ejemplo de inserción relacionada donde:

- Primero creamos la flota principal

- Luego añadimos sus misiones usando currval para mantener la relación

Consultas de Lectura (READ)

1. Consulta de Planetas con Información Completa

SELECT

p.scientific\_name,

p.common\_name,

p.population,

e.name as empire\_name,

string\_agg(DISTINCT m.name || ' (' || m.height || 'm)', ', ') as mountains

FROM Planets p

LEFT JOIN Planet\_Empires pe ON p.planet\_id = pe.planet\_id

LEFT JOIN Empires e ON pe.empire\_id = e.empire\_id

LEFT JOIN Mountains m ON p.planet\_id = m.planet\_id

GROUP BY p.planet\_id, p.scientific\_name, p.common\_name, p.population, e.name;

Esta consulta proporciona una vista completa de cada planeta, incluyendo:

- Información básica del planeta

- El imperio que lo controla (si existe)

- Una lista formateada de sus montañas

2. Consulta de Flotas y sus Capacidades

SELECT

f.galactic\_code,

f.destination,

e.name as empire\_name,

string\_agg(DISTINCT fm.mission\_type, ', ') as missions,

COUNT(DISTINCT s.ship\_id) as ship\_count

FROM Fleets f

JOIN Empires e ON f.empire\_id = e.empire\_id

LEFT JOIN Fleet\_Missions fm ON f.fleet\_id = fm.fleet\_id

LEFT JOIN Ships s ON f.fleet\_id = s.fleet\_id

GROUP BY f.fleet\_id, f.galactic\_code, f.destination, e.name;

Esta consulta muestra:

- Detalles de cada flota

- El imperio al que pertenece

- Sus misiones actuales

- La cantidad de naves asignadas

Consultas de Actualización (UPDATE)

1. Actualizar Población de un Planeta

UPDATE Planets

SET population = population + 1000000

WHERE scientific\_name = 'TER-001';

2. Reasignar una Flota a Otro Imperio

UPDATE Fleets

SET empire\_id = 2,

destination = 'Nuevo Destino'

WHERE galactic\_code = 'FLT-001';

Consultas de Eliminación (DELETE)

1. Eliminación de una Flota y sus Dependencias

-- Primero eliminamos las relaciones

DELETE FROM Fleet\_Missions

WHERE fleet\_id = (SELECT fleet\_id FROM Fleets WHERE galactic\_code = 'FLT-001');

-- Luego eliminamos la flota

DELETE FROM Fleets

WHERE galactic\_code = 'FLT-001';

2. Eliminación de un Imperio (con verificación)

-- Primero verificamos que no tenga dependencias

SELECT

COUNT(f.fleet\_id) as fleet\_count,

COUNT(pe.planet\_id) as planet\_count

FROM Empires e

LEFT JOIN Fleets f ON e.empire\_id = f.empire\_id

LEFT JOIN Planet\_Empires pe ON e.empire\_id = pe.empire\_id

WHERE e.galactic\_code = 'UFP-001'

GROUP BY e.empire\_id;

-- Si no hay dependencias, procedemos con la eliminación

DELETE FROM Empires

WHERE galactic\_code = 'UFP-001';

Estas consultas forman la base para interactuar directamente con la base de datos del sistema Star Trek. Cada una está diseñada para mantener la integridad referencial y proporcionar una operación segura y consistente sobre los datos.