Ejercicios de Consultas Geoespaciales en MongoDB

# Ejercicio 1: Crear una colección con datos geoespaciales

1. Crear una colección llamada `lugares` que almacene puntos de interés con sus coordenadas geográficas (longitud y latitud).

db.lugares.insertMany([  
 { nombre: "Parque Central", ubicacion: { type: "Point", coordinates: [-73.9654, 40.7829] } },  
 { nombre: "Museo de Historia Natural", ubicacion: { type: "Point", coordinates: [-73.9740, 40.7813] } },  
 { nombre: "Torre Eiffel", ubicacion: { type: "Point", coordinates: [2.2945, 48.8584] } },  
 { nombre: "Coliseo", ubicacion: { type: "Point", coordinates: [12.4924, 41.8902] } }  
]);

2. Crear un índice geoespacial en la colección `lugares`.

db.lugares.createIndex({ ubicacion: "2dsphere" });

# Ejercicio 2: Encontrar lugares cercanos

Encontrar todos los lugares dentro de un radio de 1 kilómetro del Parque Central (coordenadas: -73.9654, 40.7829).

db.lugares.find({  
 ubicacion: {  
 $near: {  
 $geometry: { type: "Point", coordinates: [-73.9654, 40.7829] },  
 $maxDistance: 1000  
 }  
 }  
});

# Ejercicio 3: Buscar lugares dentro de un polígono

Buscar todos los lugares que se encuentran dentro de un polígono definido por los puntos (-73.98, 40.77), (-73.97, 40.78), (-73.96, 40.78), y (-73.97, 40.77).

db.lugares.find({  
 ubicacion: {  
 $geoWithin: {  
 $geometry: {  
 type: "Polygon",  
 coordinates: [[  
 [-73.98, 40.77],  
 [-73.97, 40.78],  
 [-73.96, 40.78],  
 [-73.97, 40.77],  
 [-73.98, 40.77]  
 ]]  
 }  
 }  
 }  
});

# Ejercicio 4: Agregación geoespacial

Usar la agregación para encontrar el lugar más cercano a las coordenadas (2.2945, 48.8584) y calcular la distancia.

db.lugares.aggregate([  
 {  
 $geoNear: {  
 near: { type: "Point", coordinates: [2.2945, 48.8584] },  
 distanceField: "distancia.calculada",  
 spherical: true  
 }  
 }  
]);

# Ejercicio 5: Agrupación de lugares por proximidad

Agrupar los lugares por proximidad a un punto específico (por ejemplo, el Museo de Historia Natural) y contar cuántos lugares están dentro de cada grupo de 1 km de radio.

db.lugares.aggregate([  
 {  
 $geoNear: {  
 near: { type: "Point", coordinates: [-73.9740, 40.7813] },  
 distanceField: "distancia.calculada",  
 spherical: true  
 }  
 },  
 {  
 $bucket: {  
 groupBy: "$distancia.calculada",  
 boundaries: [0, 1000, 2000, 3000, 4000, 5000],  
 default: "Fuera de rango",  
 output: {  
 count: { $sum: 1 },  
 lugares: { $push: "$nombre" }  
 }  
 }  
 }  
]);