

Uso de Bulkwrite en MongoDB

1. Insertar múltiples documentos

Ejercicio sencillo: Inserta tres nuevos clientes en la colección con una sola operación bulkWrite.

Código:

```
db.clientes.bulkWrite([
  {
    insertOne: {
      document: {
        _id: "CL1001",
        nombre: "Carlos López",
        direccion: {
          calle: "Calle 80",
          ciudad: "Ciudad de México",
          codigoPostal: "11000"
        },
        email: "carlos.lopez@example.com",
        telefono: "+52-1-123-4567",
        fecha_registro: new Date("2023-05-01T10:00:00Z"),
        estatus: "activo",
        puntos_fidelidad: 400,
        preferencias: {
          categorias_favoritas: ["tecnología", "videojuegos"],
          notificaciones: true
        },
        metodos_pago: [
          {
            tipo: "tarjeta_credito",
            numero: "**** * 5678",
            caducidad: "06/25"
          }
        ]
      }
    }
  },
  {
    insertOne: {
      document: {
        _id: "CL1002",
        nombre: "Andrea Martínez",
        direccion: {
          calle: "Avenida Reforma",
          ciudad: "Guadalajara",
          codigoPostal: "44500"
        },
        email: "andrea.martinez@example.com",
        telefono: "+52-1-987-6543",
        fecha_registro: new Date("2023-06-15T14:30:00Z"),
        estatus: "activo",
        puntos_fidelidad: 320,
        preferencias: {
          categorias_favoritas: ["moda", "cocina"],
          notificaciones: false
        }
      }
    }
  }
])
```

```

        metodos_pago: [
          {
            tipo: "PayPal",
            correo: "andrea.martinez@paypal.com"
          }
        ]
      }
    }
  }
]
})

```

2. Actualizar múltiples documentos con condiciones

Ejercicio intermedio: Incrementa los puntos de fidelidad en 100 para todos los clientes que tengan más de 500 puntos de fidelidad.

Código:

```

db.clientes.bulkWrite([
  {
    updateMany: {
      filter: { puntos_fidelidad: { $gt: 500 } },
      update: { $inc: { puntos_fidelidad: 100 } }
    }
  }
])

```

3. Combinación de operaciones (insertar, actualizar y eliminar)

Ejercicio avanzado: Realiza las siguientes operaciones en una sola llamada:

1. Inserta un nuevo cliente.
2. Actualiza el estatus de los clientes cuyo correo electrónico termine en "example.com" a "inactivo".
3. Elimina clientes con menos de 100 puntos de fidelidad.

Código:

```

db.clientes.bulkWrite([
  {
    insertOne: {
      document: {
        _id: "CL1003",
        nombre: "Sofía Gómez",
        direccion: {
          calle: "Calle 10",
          ciudad: "Tijuana",
          codigoPostal: "22000"
        },
        email: "sofia.gomez@example.com",
        telefono: "+52-1-765-4321",
        fecha_registro: new Date("2023-07-20T09:15:00Z"),
        estatus: "activo",
        puntos_fidelidad: 150,

```

```

    preferencias: {
      categorias_favoritas: ["viajes", "fotografía"],
      notificaciones: true
    },
    metodos_pago: [
      {
        tipo: "tarjeta_credito",
        numero: "**** * 7890",
        caducidad: "10/26"
      }
    ]
  }
},
{
  updateMany: {
    filter: { email: { $regex: /@example\.com$/ } },
    update: { $set: { estatus: "inactivo" } }
  },
  {
    deleteMany: {
      filter: { puntos_fidelidad: { $lt: 100 } }
    }
  }
}
])

```

4. Usar arrayFilters para actualizar elementos específicos en un arreglo

Ejercicio avanzado con arreglos: Actualiza el campo dispositivo a "iPhone 13" en el historial de acceso para clientes cuyo último acceso fue con "iPhone".

Código:

```

db.clientes.bulkWrite([
  {
    updateMany: {
      filter: { "historial_acceso.dispositivo": "iPhone" },
      update: {
        $set: { "historial_acceso.$[elem].dispositivo": "iPhone 13" }
      },
      arrayFilters: [
        { "elem.dispositivo": "iPhone" }
      ]
    }
  }
])

```

5. Gestionar un programa de fidelidad

Enunciado: Realiza las siguientes operaciones:

1. Inserta dos nuevos clientes con categorías favoritas distintas.
2. Incrementa en 50 puntos los puntos de fidelidad de los clientes activos que ya tienen más de 300 puntos.
3. Cambia el estatus de los clientes con menos de 200 puntos de fidelidad a "inactivo".
4. Elimina todos los clientes que no tienen métodos de pago registrados.

Código:

```
db.clientes.bulkWrite([
  {
    insertOne: {
      document: {
        _id: "CL1004",
        nombre: "Juan Pérez",
        direccion: {
          calle: "Calle 20",
          ciudad: "Toluca",
          codigoPostal: "50120"
        },
        email: "juan.perez@example.com",
        telefono: "+52-1-888-5555",
        fecha_registro: new Date("2023-09-01T08:00:00Z"),
        estatus: "activo",
        puntos_fidelidad: 300,
        preferencias: {
          categorias_favoritas: ["hogar", "jardinería"],
          notificaciones: true
        },
        metodos_pago: []
      }
    }
  },
  {
    insertOne: {
      document: {
        _id: "CL1005",
        nombre: "Luisa Morales",
        direccion: {
          calle: "Calle 50",
          ciudad: "Cancún",
          codigoPostal: "77500"
        },
        email: "luisa.morales@example.com",
        telefono: "+52-1-999-4444",
        fecha_registro: new Date("2023-08-15T12:00:00Z"),
        estatus: "activo",
        puntos_fidelidad: 150,
        preferencias: {
          categorias_favoritas: ["moda", "belleza"],
          notificaciones: false
        },
        metodos_pago: [
          {

```

```

        tipo: "tarjeta_credito",
        numero: "**** * 5678",
        caducidad: "11/25"
      }
    ]
  },
  {
    updateMany: {
      filter: { estatus: "activo", puntos_fidelidad: { $gt: 300 } },
      update: { $inc: { puntos_fidelidad: 50 } }
    },
    {
      updateMany: {
        filter: { puntos_fidelidad: { $lt: 200 } },
        update: { $set: { estatus: "inactivo" } }
      },
      {
        deleteMany: {
          filter: { metodos_pago: { $size: 0 } }
        }
      }
    }
  ]
})

```

6. Operaciones en arreglos complejos

Enunciado: Realiza las siguientes tareas:

1. Inserta un cliente nuevo con un historial de acceso inicial.
2. Actualiza el último acceso de los clientes que han usado un iPhone para que refleje un acceso desde "iPhone 15 Pro".
3. Elimina el historial de acceso de los clientes con más de 3 accesos.

Código:

```

db.clientes.bulkWrite([
  {
    insertOne: {
      document: {
        _id: "CL1010",
        nombre: "Mariana Torres",
        direccion: {
          calle: "Calle 100",
          ciudad: "Puebla",
          codigoPostal: "72500"
        },
        email: "mariana.torres@example.com",
        telefono: "+52-1-777-3333",
        fecha_registro: new Date("2023-10-10T09:00:00Z"),
        estatus: "activo",
        puntos_fidelidad: 250,
        historial_acceso: [
          { fecha: new Date("2024-01-01T10:00:00Z"), dispositivo: "Laptop" }
        ],
      }
    }
  }
])

```

```

        preferencias: {
            categorias_favoritas: ["tecnología", "viajes"],
            notificaciones: true
        },
        metodos_pago: []
    }
},
{
    updateMany: {
        filter: { "historial_acceso.dispositivo": "iPhone" },
        update: {
            $set: { "historial_acceso.$[elem].dispositivo": "iPhone 15 Pro" }
        },
        arrayFilters: [{ "elem.dispositivo": "iPhone" }]
    },
    deleteMany: {
        filter: { $expr: { $gt: [{ $size: "$historial_acceso" }, 3] } }
    }
}
])

```

Este ejercicio arroja un error, ¿a qué se debe dicho error?

¿Con esta nueva versión, por qué no ocurre el error?

```

db.clientes.bulkWrite([
    {
        insertOne: {
            document: {
                _id: "CL1010",
                nombre: "Mariana Torres",
                direccion: {
                    calle: "Calle 100",
                    ciudad: "Puebla",
                    codigoPostal: "72500"
                },
                email: "mariana.torres@example.com",
                telefono: "+52-1-777-3333",
                fecha_registro: new Date("2023-10-10T09:00:00Z"),
                estatus: "activo",
                puntos_fidelidad: 250,
                historial_acceso: [
                    { fecha: new Date("2024-01-01T10:00:00Z"), dispositivo: "Laptop" }
                ],
                preferencias: {
                    categorias_favoritas: ["tecnología", "viajes"],
                    notificaciones: true
                },
                metodos_pago: []
            }
        },
        updateMany: {
            filter: { "historial_acceso.dispositivo": "iPhone" },
            update: {
                $set: { "historial_acceso.$[elem].dispositivo": "iPhone 15 Pro" }
            },
        },
    },
    {
        updateMany: {
            filter: { "historial_acceso.dispositivo": "iPhone" },
            update: {
                $set: { "historial_acceso.$[elem].dispositivo": "iPhone 15 Pro" }
            },
        },
    },
]
)

```

```

        arrayFilters: [{ "elem.dispositivo": "iPhone" }]
    },
    {
        deleteMany: {
            filter: {
                historial_acceso: { $exists: true, $type: "array" },
                $expr: { $gt: [{ $size: "$historial_acceso" }, 3] }
            }
        }
    }
}
])

```

7. Gestionar métodos de pago y notificaciones

Enunciado: Realiza las siguientes tareas:

1. Agrega un método de pago PayPal a los clientes que no tienen ninguno.
2. Desactiva las notificaciones de los clientes que tienen más de 2 métodos de pago registrados.
3. Elimina los métodos de pago vencidos en diciembre de 2024.

Código:

```

db.clientes.bulkWrite([
  {
    updateMany: {
      filter: { metodos_pago: { $size: 0 } },
      update: {
        $push: { metodos_pago: { tipo: "PayPal", correo: "nuevo@paypal.com" } }
      }
    }
  },
  {
    updateMany: {
      filter: { $expr: { $gt: [{ $size: "$metodos_pago" }, 2] } },
      update: { $set: { "preferencias.notificaciones": false } }
    }
  },
  {
    updateMany: {
      filter: { "metodos_pago.caducidad": "12/24" },
      update: {
        $pull: { metodos_pago: { caducidad: "12/24" } }
      }
    }
  }
])

```

8. Limpieza y normalización de datos

Enunciado: Realiza las siguientes tareas:

1. Cambia el estatus de los clientes con más de 1000 puntos de fidelidad a "premium".
2. Elimina a los clientes con direcciones incompletas (sin ciudad o codigoPostal).
3. Actualiza todos los números de teléfono para que incluyan un prefijo "+52".

Código:

```
db.clientes.bulkWrite([
  {
    updateMany: {
      filter: { puntos_fidelidad: { $gt: 1000 } },
      update: { $set: { estatus: "premium" } }
    }
  },
  {
    deleteMany: {
      filter: {
        $or: [
          { "direccion.ciudad": { $exists: false } },
          { "direccion.codigoPostal": { $exists: false } }
        ]
      }
    }
  },
  {
    updateMany: {
      filter: { telefono: { $not: { $regex: /^\\+52-/ } } },
      update: { $set: { telefono: { $concat: ["+52-", "$telefono"] } } }
    }
  }
])
```

9. Complejo: Administración y auditoría

Enunciado: Realiza las siguientes tareas:

1. Inserta dos nuevos clientes premium con métodos de pago múltiples.
2. Incrementa en un 10% los puntos de fidelidad de los clientes premium.
3. Elimina clientes que no hayan registrado actividad en el último año.
4. Cambia el correo electrónico a "actualizado@example.com" para clientes con correos duplicados.

Código:

```
db.clientes.bulkWrite([
  {
    insertOne: {
      document: {
        _id: "CL7001",
        nombre: "Ricardo Sánchez",
        direccion: {
          calle: "Calle 150",
          ciudad: "Querétaro",
          codigoPostal: "76000"
        }
      }
    }
  },
  {
    insertOne: {
      document: {
        _id: "CL7002",
        nombre: "María González",
        direccion: {
          calle: "Calle 100",
          ciudad: "Querétaro",
          codigoPostal: "76000"
        }
      }
    }
  }
])
```



```

        email: "ricardo.sanchez@example.com",
        telefono: "+52-1-555-1212",
        fecha_registro: new Date("2024-01-01T10:00:00Z"),
        estatus: "premium",
        puntos_fidelidad: 1200,
        metodos_pago: [
            { tipo: "tarjeta_credito", numero: "**** * 4321", caducidad:
"10/25" },
            { tipo: "PayPal", correo: "ricardo@paypal.com" }
        ]
    },
    {
        insertOne: {
            document: {
                _id: "CL8001",
                nombre: "Ana López",
                direccion: {
                    calle: "Calle 200",
                    ciudad: "León",
                    codigoPostal: "37100"
                },
                email: "ana.lopez@example.com",
                telefono: "+52-1-222-8888",
                fecha_registro: new Date("2024-02-01T12:00:00Z"),
                estatus: "premium",
                puntos_fidelidad: 1500,
                metodos_pago: [
                    { tipo: "tarjeta_credito", numero: "**** * 9876", caducidad:
"09/26" }
                ]
            }
        },
        {
            updateMany: {
                filter: { estatus: "premium" },
                update: { $mul: { puntos_fidelidad: 1.1 } }
            }
        },
        {
            deleteMany: {
                filter: { "historial_acceso.fecha": { $lt: new Date("2023-01-
01T00:00:00Z") } }
            }
        },
        {
            updateMany: {
                filter: { email: { $in: ["duplicado@example.com"] } },
                update: { $set: { email: "actualizado@example.com" } }
            }
        }
    ]
})

```