

# Practica MongoDB

Jaime Jiménez Ardanaz

## Modelado de Datos

Para empezar, diseño la base de datos empleando 3 colecciones: **productos**, **empleados** y **clientes**. La colección proveedores estará incrustada dentro de productos, mientras que la colección pedidos estará incrustada dentro de clientes. A su vez, dentro de clientes se encuentra un array de clasificación que funcionará como sistema de valoración de un o varios productos.

## Operaciones CRUD de Mongo

Esta es la forma en la que he generado la base de datos.

```
db.productos.insertMany([
  {
    nombre: "Caramelos de limón", precio: 2,
    descripción: "Bolsa de 250 gramos de caramelos de limón.",
    categoría: "Caramelos", stock: 20, proveedor: "Proveedores Edores"
  },
  {
    nombre: "Pack de regalices", precio: 0.60,
    descripción: "Bolsa de 4 regalices de cereza.",
    categoría: "Gominolas", stock: 12, proveedor: "La Andaluza"
  },
  {
    nombre: "Refresco de uva", precio: 1.80,
    descripción: "Lata que contiene una bebida carbonatada con sabor a uva.",
    categoría: "Refrescos", stock: 6, proveedor: "Proveedores Edores"
  },
  {
    nombre: "Caramelos de menta", precio: 2,
    descripción: "Bolsa de 250 gramos de caramelos de menta.",
    categoría: "Caramelos", stock: 0, proveedor: "Productos Hierbabuena"
  },
  {
    nombre: "Tableta de chocolate", precio: 1.80,
    descripción: "200 gramos de chocolate templado y enfriado convenientemente en un molde.",
    categoría: "Chocolate", stock: 6, proveedor: "Fabrica de M. M."
  },
  {
    nombre: "Bolsa de cacahuets", precio: 2.50,
    descripción: "Que caros están los frutos secos.",
    categoría: "Frutos Secos", stock: 3, proveedor: "Productos Hierbabuena"
  },
  {
    nombre: "Refresco de mora", precio: 1.75,
    descripción: "Lata que contiene una bebida carbonatada con sabor a mora.",
    categoría: "Refrescos", stock: 40, proveedor: "Estaciones"
  },
  {
    nombre: "Moras", precio: 0.10,
    descripción: "1 mora de gominola.",
    categoría: "Gominolas", stock: 3, proveedor: "La Andaluza"
  },
  {
    nombre: "Caramelos de regaliz", precio: 2,
    descripción: "Bolsa de 250 gramos de caramelos de regaliz.",
    categoría: "Caramelos", stock: 12, proveedor: "La Andaluza"
  },
  {
    nombre: "Piruleta de Fresa", precio: 1.00,
    descripción: "Piruleta redonda, de color rojo y blanco, con sabos a fresa.",
    categoría: "Piruletas", stock: 6, proveedor: "Minas Durin"
  }
])
```

```
db.empleados.insertMany([
  {
    nombre: "Peter P. Peterson Sr.", departamento: 'Departamento de Publicidad',
    salario: 350, fecha_contratacion: ISODate("2007-05-19")},
  {
    nombre: "Frank Colina", departamento: 'Departamento de Reparto',
    salario: 500, fecha_contratacion: ISODate("2014-06-11")},
  {
    nombre: "Pepe Gurbindo", departamento: 'Departamento de Ventas',
    salario: 650, fecha_contratacion: ISODate("2019-12-31")},
  {
    nombre: "Mariano Jarroy", departamento: 'Departamento de Logística',
    salario: 800, fecha_contratacion: ISODate("2020-11-11")},
  {
    nombre: "Nuria García", departamento: 'Departamento de Ventas',
    salario: 720, fecha_contratacion: ISODate("2018-05-22")},
  {
    nombre: "Pablo Montesano", departamento: 'Departamento de Finanzas',
    salario: 600, fecha_contratacion: ISODate("2020-09-01")},
  {
    nombre: "Juan Tena", departamento: 'Departamento de Recursos Humanos',
    salario: 600, fecha_contratacion: ISODate("2013-08-20")},
  {
    nombre: "Raul Lopez", departamento: 'Departamento de Legal',
    salario: 650, fecha_contratacion: ISODate("2021-05-25")},
  {
    nombre: "Marcelo Navarro", departamento: 'Departamento de Atención al Cliente',
    salario: 550, fecha_contratacion: ISODate("2020-07-21")},
  {
    nombre: "Patricia Reyes", departamento: 'Departamento de Reparto',
    salario: 800, fecha_contratacion: ISODate("2021-05-04")}
])
```

```
db.clientes.insertMany([
  {
    nombre: "Peter P. Peterson", correo: 'petersonjr@gmail.com',
    telefono: 663775123, direccion: "c/ Huerto Abierto, 85",
    clasificacion: [(rating: 5, producto: ObjectId("XXXXXXXXXXXX"))],
    pedidos: [
      {
        (producto: ObjectId("XXXXXXXXXXXX"), fecha: ISODate("2024-09-09"))
        (producto: ObjectId("XXXXXXXXXXXX"), fecha: ISODate("2024-07-28"))
        (producto: ObjectId("XXXXXXXXXXXX"), fecha: ISODate("2024-09-13"))
      }
    ],
    nombre: "Alberto Gutierrez", correo: 'gutiilabert@hotmail.com',
    telefono: 6437549545, direccion: "av. Campo Campanario, 12 548",
    nombre: "Marta Latorre", correo: 'martalatorre@gmail.es',
    telefono: 657283924, direccion: "c/ Torremolinos, 89",
    clasificacion: [
      (rating: 2, producto: ObjectId("XXXXXXXXXXXX")),
      (rating: 4, producto: ObjectId("XXXXXXXXXXXX"))
    ],
    pedidos: [
      (producto: ObjectId("XXXXXXXXXXXX"), fecha: ISODate("2024-01-09"))
      (producto: ObjectId("XXXXXXXXXXXX"), fecha: ISODate("2024-10-29"))
    ]
  },
  {
    nombre: "Gerardo Masana", correo: 'gerardomasana@outlook.org',
    telefono: 654343233, direccion: "c/ Argentina Republicana, 6",
    clasificacion: [(rating: 3, producto: ObjectId("XXXXXXXXXXXX"))]
  },
  {
    nombre: "Godofredo Reyes", correo: 'reydegodo@hotmail.com',
    telefono: 634312122, direccion: "av. Burgueses, 5 156",
    clasificacion: [
      (rating: 1, producto: ObjectId("XXXXXXXXXXXX")),
      (rating: 2, producto: ObjectId("XXXXXXXXXXXX")),
      (rating: 1, producto: ObjectId("XXXXXXXXXXXX")),
      (rating: 3, producto: ObjectId("XXXXXXXXXXXX"))
    ],
    pedidos: [(producto: ObjectId("XXXXXXXXXXXX"), fecha: ISODate("2020-12-01"))]
  },
  {
    nombre: "Paloma Herman", correo: 'palomaherfer@gmail.com',
    telefono: 659989991, direccion: "c/ Avenida de la Mar, 12",
    clasificacion: [(rating: 3, producto: ObjectId("XXXXXXXXXXXX"))]
  }
])
```

```
}, {
  nombre: "Ricardo Itxard", correo: 'itxric13@gmail.com',
  telefono: 656565121, direccion: "Olvr. Galicia , 18 34 Izq",
  pedidos: [(producto: ObjectId("XXXXXXXXXXXX"), fecha: ISODate("2007-05-19"))]
}, {
  nombre: "Malfredi I. Jhonson", correo: 'pinguin9@gmail.com',
  telefono: 689911239, direccion: "c/ Polar Glacial, 2",
  clasificacion: [
    (rating: 5, producto: ObjectId("XXXXXXXXXXXX")),
    (rating: 5, producto: ObjectId("XXXXXXXXXXXX")),
    (rating: 5, producto: ObjectId("XXXXXXXXXXXX"))
  ]
}, {
  nombre: "Matias", correo: 'matiascas@gmail.com',
  telefono: 65466562, direccion: "av. Enida, 8 24C",
  clasificacion: [(rating: 5, producto: ObjectId("XXXXXXXXXXXX"))],
  pedidos: [
    (producto: ObjectId("XXXXXXXXXXXX"), fecha: ISODate("2023-11-23"))
    (producto: ObjectId("XXXXXXXXXXXX"), fecha: ISODate("2024-09-24"))
    (producto: ObjectId("XXXXXXXXXXXX"), fecha: ISODate("2024-07-30"))
    (producto: ObjectId("XXXXXXXXXXXX"), fecha: ISODate("2024-09-10"))
  ]
}, {
  nombre: "Juan Antonio Suler", correo: 'suler74@hotmail.com',
  telefono: 651811990, direccion: "c/ Parque de San Cristobal",
  clasificacion: [(rating: 2, producto: ObjectId("XXXXXXXXXXXX"))]
}, {
  nombre: "Drake Mallard", correo: 'drakenoduck@gmail.com',
  telefono: 633190872, direccion: "c/ Opera, 23",
  pedidos: [
    (producto: ObjectId("XXXXXXXXXXXX"), fecha: ISODate("2024-06-23"))
    (producto: ObjectId("XXXXXXXXXXXX"), fecha: ISODate("2023-05-19"))
  ]
}, {
  nombre: "Antonio Ramirez", correo: 'antoniram@gmail.com',
  telefono: 665881923, direccion: "c/ Plaza Londres, 87"
}
```

Actualización de datos dentro de las colecciones:

```
db.productos.updateOne(
  {nombre: "Caramelos de regaliz"},
  {$set: { nombre: "Caramelos de regaliz negro" }}
);

db.clientes.updateOne(
  {nombre: "Pepe Gurbindo"},
  {$set: { telefono: 677199231 }}
);

db.empleados.updateOne(
  {nombre: "Pepe Gurbindo"},
  {$set: { nombre: "Patxi Gurbindo" }}
);
```

```
database> db.productos.updateOne(
... {nombre: "Caramelos de regaliz"},
... {$set: { nombre: "Caramelos de regaliz negro" }}
... );
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

```
database> db.clientes.updateOne(
... {nombre: "Antonio Ramirez"},
... {$set: { telefono: 677199231 }}
... );
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

```
database> db.empleados.updateOne(
... {nombre: "Pepe Gurbindo"},
... {$set: { nombre: "Patxi Gurbindo" }}
... );
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

Eliminar registros de empleados y clientes:

```
db.clientes.deleteOne({ nombre: "Antonio Ramirez" });
db.empleados.deleteOne({ nombre: "Marcelo Navarro" });
```

```
database> db.clientes.deleteOne({ nombre: "Antonio Ramirez" });
{ acknowledged: true, deletedCount: 1 }
```

```
database> db.empleados.deleteOne({ nombre: "Marcelo Navarro" });
{ acknowledged: true, deletedCount: 1 }
```

## Agreggation pipelines

### 1. Encuentra el producto más vendido en la tienda.

```
db.clientes.aggregate([
  { $unwind: "$pedidos" },
  { $group: { _id: "$pedidos.producto", num_pedido: { $sum: 1 } } },
  { $sort: { num_pedido: -1 } }, { $limit: 1 }
]);
```

```
database> db.clientes.aggregate(
...   [
...     { $unwind: "$pedidos" },
...     { $group: { _id: "$pedidos.producto", num_pedido: { $sum: 1 } } },
...     { $sort: { num_pedido: -1 } }, { $limit: 1 }
...   ]
... );
[ { _id: ObjectId('673f7b902d11d42468fe6913'), num_pedido: 2 } ]
database>
```

### 2. Identifica a los clientes que han realizado más de 3 pedidos en el último mes.

```
db.clientes.aggregate([
  { $unwind: "$pedidos" },
  { $group: { _id: "$nombre", num_pedido: { $sum: 1 } } },
  { $match: { "pedidos.fecha": { $gte: new Date(new Date().setMonth(new Date().getMonth()-1)) } } },
  { $match: { num_pedido: { $gt: 3 } } }
]);
```

```
database> db.clientes.aggregate([
...   { $unwind: "$pedidos" },
...   { $group: { _id: "$nombre", num_pedido: { $sum: 1 } } },
...   { $match: { "pedidos.fecha": { $gte: new Date(new Date().setMonth(new Date().getMonth() - 1)) } } },
...   { $match: { num_pedido: { $gt: 3 } } }
... ]);
database>
```

En este caso no había ninguno.

### 3. Calcula la ganancia total por empleado en los últimos 6 meses.

```
db.empleados.aggregate([
  { $match: { "fecha_contratacion": { $lte: new Date(new Date().setMonth(new Date().getMonth()-6)) } } },
  { $project: { _id: 0, nombre: 1, total_salario: { $multiply: ["$salario", 6] } } }
]);
```

```
database> db.empleados.aggregate([
...   { $match: { "fecha_contratacion": { $lte: new Date(new Date().setMonth(new Date().getMonth()-6)) } } },
...   { $project: { _id: 0, nombre: 1, total_salario: { $multiply: ["$salario", 6] } } }
... ]);
[
  { nombre: 'Peter P. Peterson Sr.', total_salario: 2100 },
  { nombre: 'Frank Colina', total_salario: 3000 },
  { nombre: 'Pepe Gurbindo', total_salario: 3900 },
  { nombre: 'Mariano Jarroy', total_salario: 4800 },
  { nombre: 'Nuria Garcia', total_salario: 4320 },
  { nombre: 'Pablo Montesanto', total_salario: 3600 },
  { nombre: 'Juan Tena', total_salario: 3600 },
  { nombre: 'Raul Lopez', total_salario: 3900 },
  { nombre: 'Patricia Reyes', total_salario: 4800 },
  { nombre: 'Marcelo Navarro', total_salario: 3300 }
]
```

#### 4. Encuentra el promedio de precios de productos por categoría.

```
db.productos.aggregate([
  {$group: {_id: "$categoria", media: { $avg: "$precio" } }}
]);
```

```
database> db.productos.aggregate([
...   {$group: {_id: "$categoria", media: { $avg: "$precio" } }}
... ]);
[
  { _id: 'Piruletas', media: 1.6 },
  { _id: 'Refrescos', media: 1.775 },
  { _id: 'Caramelos', media: 2 },
  { _id: 'Chocolate', media: 1.8 },
  { _id: 'Frutos Secos', media: 2.5 },
  { _id: 'Gominolas', media: 0.35 }
]
```

#### 5. Agrupa los productos por proveedor y muestra el recuento de productos por proveedor.

```
db.productos.aggregate([
  {$group: {_id: "$proveedor", num_productos: { $sum: 1 } }}
]);
```

```
database> db.productos.aggregate([
...   {$group: {_id: "$proveedor", num_productos: { $sum: 1 } }}
... ]);
[
  { _id: 'Provedores Edores', num_productos: 2 },
  { _id: 'La Andaluza', num_productos: 3 },
  { _id: 'Fabrica de W. W.', num_productos: 1 },
  { _id: 'Estaciones', num_productos: 1 },
  { _id: 'Productos Hierbabuena', num_productos: 2 },
  { _id: 'Minas Durin', num_productos: 1 }
]
```

#### 6. Encuentra los productos que han estado fuera de stock por más de 30 días.

```
db.clientes.aggregate([
  {$unwind: "$pedidos"},
  {
    $lookup: {
      from: "productos",
      localField: "pedidos.producto",
      foreignField: "_id",
      as: "pedidos_realizados"
    }
  },
  {$match: {"pedidos_realizados.stock": {$eq: 0}}},
  {$limit: 1},
  {$match: {"pedidos.fecha": { $gte: new Date(new Date().setMonth(new Date().getMonth() - 6))}}},
  {$project: { _id: 0, "pedidos_realizados.nombre": 1}}
]);
```

```
database> db.clientes.aggregate([
...   {$unwind: "$pedidos"},
...   {
...     $lookup: {
...       from: "productos",
...       localField: "pedidos.producto",
...       foreignField: "_id",
...       as: "pedidos_realizados"
...     }
...   },
...   {$match: {"pedidos_realizados.stock": {$eq: 0}}},
...   {$limit: 1},
...   {$match: {"pedidos.fecha": { $gte: new Date(new Date().setMonth(new Date().getMonth() - 6))}}},
...   {$project: { _id: 0, "pedidos_realizados.nombre": 1}}
... ]);
database>
```

## 7. Identifica a los clientes que gastaron más de un cierto monto en pedidos el último trimestre.

```
db.clientes.aggregate([
  {$unwind: "$pedidos"},
  {
    $lookup: {
      from: "productos",
      localField: "pedidos.producto",
      foreignField: "_id",
      as: "producto_info"
    }
  },
  {$unwind: "$producto_info"},
  {$match: { "pedidos.fecha": { $gte: new Date(new Date().setMonth(new Date().getMonth() - 3)) } }},
  {$group: { _id: "$nombre", gasto: { $sum: "$producto_info.precio" } }},
  {$match: { "gasto": { $gte: 3 } }},
  {$project: { _id: 0, nombre: "$_id", gasto: 1 }}
]);
```

```
database> db.clientes.aggregate([
...   {$unwind: "$pedidos"},
...   {
...     $lookup: {
...       from: "productos",
...       localField: "pedidos.producto",
...       foreignField: "_id",
...       as: "producto_info"
...     }
...   },
...   {$unwind: "$producto_info"},
...   {$match: { "pedidos.fecha": { $gte: new Date(new Date().setMonth(new Date().getMonth() - 3)) } }},
...   {$group: { _id: "$nombre", gasto: { $sum: "$producto_info.precio" } }},
...   {$match: { "gasto": { $gte: 3 } }},
...   {$project: { _id: 0, nombre: "$_id", gasto: 1 }}
... ]);
[
  { gasto: 3.8, nombre: 'Matias' },
  { gasto: 3.75, nombre: 'Peter P. Peterson' }
]
```

## 8. Encuentra los productos más populares basados en las calificaciones de los clientes.

```
db.clientes.aggregate([
  {$unwind: "$clasificacion"},
  {
    $lookup: {
      from: "productos",
      localField: "clasificacion.producto",
      foreignField: "_id",
      as: "producto_info"
    }
  },
  {$unwind: "$producto_info"},
  {$group: { _id: "$producto_info.nombre", media: { $avg: "$clasificacion.rating" } }},
  {$sort: { media: -1 }},
  {$limit: 5}
]);
```

```
database> db.clientes.aggregate([
...   {$unwind: "$clasificacion"},
...   {
...     $lookup: {
...       from: "productos",
...       localField: "clasificacion.producto",
...       foreignField: "_id",
...       as: "producto_info"
...     }
...   },
...   {$unwind: "$producto_info"},
...   {$group: { _id: "$producto_info.nombre", media: { $avg: "$clasificacion.rating" } }},
...   {$sort: { media: -1 }},
...   {$limit: 5}
... ]);
[
  { _id: 'Caramelos de regalo', media: 5 },
  { _id: 'Moras', media: 4 },
  { _id: 'Refresco de mora', media: 4 },
  { _id: 'Pack de regalices', media: 3.3333333333333335 },
  { _id: 'Refresco de uva', media: 3 }
]
database>
```

9. Realiza una agregación que muestre el salario promedio de los empleados por departamento.

```
db.empleados.aggregate([
  { $group: { _id: "$departamento", media: { $avg: '$salario' } } }
]);

database> db.empleados.aggregate([
...   { $group: { _id: "$departamento", media: { $avg: '$salario' } } }
... ]);
[
  { _id: 'Departamento de Recursos Humanos', media: 600 },
  { _id: 'Departamento de Reparto', media: 650 },
  { _id: 'Departamento de Publicidad', media: 350 },
  { _id: 'Departamento de Ventas', media: 685 },
  { _id: 'Departamento de Atencion al Cliente', media: 550 },
  { _id: 'Departamento de Logística', media: 800 },
  { _id: 'Departamento de Legal', media: 650 },
  { _id: 'Departamento de Finanzas', media: 600 }
]
```

## Índices

1. Crea un índice simple en la colección "productos" para acelerar las búsquedas por nombre del producto.

```
database> db.productos.createIndex({ nombre: 1 });
nombre_1
```

2. Implementa un índice compuesto en la colección "pedidos" que incluya el cliente y la fecha del pedido.

```
database> db.pedidos.createIndex({ cliente: 1, fecha: 1 });
cliente_1_fecha_1
database>
```

3. Crea un índice de texto en la colección "productos" para habilitar búsquedas de texto completo en la descripción de los productos.

```
database> db.productos.createIndex({ descripcion: "text" });
descripcion_text
```