

## MongoDB Query Report – Aimee Final DB

### 1. InsertOne – Add a New Vendor

Used to insert a single vendor record into the collection.

Expected Output:

```
> db.vendors.insertOne({
  _id: 11,
  name: "Tina Lopez",
  phone: "647-555-2222",
  email: "tina@example.com",
  startDate: ISODate("2025-03-29"),
  isLocal: true
})
< {
  acknowledged: true,
  insertedId: 11
}
```

### 2. InsertMany – Add Multiple Sales

Used to insert multiple documents at once.

Expected Output:

```
> db.sales.insertMany([
  { _id: 1011, productId: 105, saleDate: ISODate("2025-03-29"), quantity: 2 },
  { _id: 1012, productId: 109, saleDate: ISODate("2025-03-29"), quantity: 5 }
])
< {
  acknowledged: true,
  insertedIds: {
    '0': 1011,
    '1': 1012
  }
}
```

### 3. Find – All Local Vendors

Filters vendors by a boolean field `isLocal`.

**Expected Output:**

```
> db.vendors.find({ isLocal: true })
< {
  _id: 1,
  name: 'Aimee Carter',
  phone: '416-555-1234',
  email: 'aimee@example.com',
  startDate: 2023-04-01T00:00:00.000Z,
  isLocal: true
}
{
  _id: 3,
  name: 'Lina Moore',
  phone: '905-555-2323',
  email: 'lina@example.com',
  startDate: 2023-11-10T00:00:00.000Z,
  isLocal: true
}
{
  _id: 5,
  name: 'Sophia Lin',
  phone: '613-555-8787',
  email: 'sophia@example.com',
  startDate: 2022-08-22T00:00:00.000Z,
  isLocal: true
}
{
  _id: 6,
  name: 'Noah Ray',
  phone: '905-555-3344',
  email: 'noah@example.com',
  startDate: 2024-02-05T00:00:00.000Z,
  isLocal: true
}
```

```
}  
{  
  _id: 8,  
  name: 'Felix Kim',  
  phone: '647-555-2233',  
  email: 'felix@example.com',  
  startDate: 2024-04-10T00:00:00.000Z,  
  isLocal: true  
}  
{  
  _id: 9,  
  name: 'Rhea Patel',  
  phone: '613-555-1029',  
  email: 'rhea@example.com',  
  startDate: 2022-12-01T00:00:00.000Z,  
  isLocal: true  
}  
{  
  _id: 11,  
  name: 'Tina Lopez',  
  phone: '647-555-2222',  
  email: 'tina@example.com',  
  startDate: 2025-03-29T00:00:00.000Z,  
  isLocal: true  
}
```

**4. Find+ – Vendor Names and Emails Only**  
Returns only specific fields.

**Expected Output:**

```
> db.vendors.find({}, { name: 1, email: 1, _id: 0 })
< {
  name: 'Aimee Carter',
  email: 'aimee@example.com'
}
{
  name: 'David Stone',
  email: 'david@example.com'
}
{
  name: 'Lina Moore',
  email: 'lina@example.com'
}
{
  name: 'Jared Beck',
  email: 'jared@example.com'
}
{
  name: 'Sophia Lin',
  email: 'sophia@example.com'
}
{
  name: 'Noah Ray',
  email: 'noah@example.com'
}
{
  name: 'Clara Wells',
  email: 'clara@example.com'
}
{
  name: 'Felix Kim',
  email: 'felix@example.com'
}
```

```
{
  name: 'Rhea Patel',
  email: 'rhea@example.com'
}
{
  name: 'Leo Zhang',
  email: 'leo@example.com'
}
{
  name: 'Tina Lopez',
  email: 'tina@example.com'
}
```

#### 5. UpdateOne – Modify a Single Vendor

To change specific details (like contact info) of one vendor.

Expected Output:

```
> db.vendors.updateOne(
  { _id: 2 },
  { $set: { phone: "647-555-8899" } }
)
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

#### 6. UpdateMany – Flag Legacy Vendors

To identify vendors active before 2023.

Expected Output:

```
> db.vendors.updateMany(  
  { startDate: { $lt: ISODate("2023-01-01") } },  
  { $set: { legacyVendor: true } }  
)  
< {  
  acknowledged: true,  
  insertedId: null,  
  matchedCount: 2,  
  modifiedCount: 2,  
  upsertedCount: 0  
}
```

#### 7. DeleteOne – Remove a Specific Product

Used to remove a discontinued or duplicate product.

Expected Output:

```
> db.products.deleteOne({ _id: 110 })  
< {  
  acknowledged: true,  
  deletedCount: 1  
}
```

#### 8. DeleteMany – Clean Up Out-of-Stock Products

Maintains a clean catalog by removing unavailable items.

Expected Output:

```
> db.products.deleteMany({ inStock: { $lte: 0 } })  
< {  
  acknowledged: true,  
  deletedCount: 0  
}
```



## 9. Aggregation – Total Quantity Sold Per Product

Helps identify best-selling products.

Expected Output:

```
> db.sales.aggregate([
  { $group: { _id: "$productId", totalSold: { $sum: "$quantity" } } },
  { $sort: { totalSold: -1 } }
])
< {
  _id: 109,
  totalSold: 8
}
{
  _id: 107,
  totalSold: 5
}
{
  _id: 104,
  totalSold: 3
}
{
  _id: 105,
  totalSold: 3
}
{
  _id: 101,
  totalSold: 2
}
{
  _id: 106,
  totalSold: 2
}
{
  _id: 110,
  totalSold: 1
}
}
```

```
{
  _id: 102,
  totalSold: 1
}
{
  _id: 103,
  totalSold: 1
}
{
  _id: 108,
  totalSold: 1
}
}
```

## 10. Aggregation – Count of Products per Vendor

Understand product contributions by vendor.

Expected Output:.

```
> db.products.aggregate([
  { $group: { _id: "$vendorId", count: { $sum: 1 } } },
  { $sort: { count: -1 } }
])
< {
  _id: 1,
  count: 1
}
{
  _id: 3,
  count: 1
}
{
  _id: 6,
  count: 1
}
{
  _id: 2,
  count: 1
}
{
  _id: 8,
  count: 1
}
{
  _id: 4,
  count: 1
}
{
  _id: 5,
  count: 1
}
}
```

```
{
  _id: 9,
  count: 1
}
{
  _id: 7,
  count: 1
}
```

## 11. Sort – Products by Price

Lets users browse items from highest to lowest price.

Expected Output:.

```
> db.products.find().sort({ price: -1 })
< {
  _id: 103,
  vendorId: 3,
  name: 'Steel Hunting Knife',
  category: 'Knife',
  material: 'Carbon Steel',
  price: 45,
  inStock: 6
}
{
  _id: 105,
  vendorId: 5,
  name: 'Polished Amethyst Cluster',
  category: 'Crystal',
  material: 'Amethyst',
  price: 35,
  inStock: 5
}
{
  _id: 106,
  vendorId: 6,
  name: 'Hand-Carved Wooden Coaster Set',
  category: 'Home Decor',
  material: 'Oak',
  price: 22,
  inStock: 8
}
```

```
{
  _id: 104,
  vendorId: 4,
  name: 'Coin Pouch with Embroidery',
  category: 'Pouch',
  material: 'Leather',
  price: 20,
  inStock: 10
}
{
  _id: 108,
  vendorId: 8,
  name: 'Beaded Leather Bracelet',
  category: 'Jewelry',
  material: 'Leather & Beads',
  price: 18,
  inStock: 15
}
{
  _id: 102,
  vendorId: 2,
  name: 'Crystal Sun Catcher',
  category: 'Crystal',
  material: 'Quartz',
  price: 15,
  inStock: 12
}
```

```

{
  _id: 107,
  vendorId: 7,
  name: 'Decorative Resin Keychain',
  category: 'Keychain',
  material: 'Resin',
  price: 10,
  inStock: 20
}
{
  _id: 101,
  vendorId: 1,
  name: 'Leather Corner Bookmark',
  category: 'Bookmark',
  material: 'Leather',
  price: 8,
  inStock: 25
}
{
  _id: 109,
  vendorId: 9,
  name: 'Tumbled Rose Quartz',
  category: 'Crystal',
  material: 'Rose Quartz',
  price: 6,
  inStock: 30
}

```

## 12. Indexing – Improve Lookup on vendorId

Boosts performance when filtering products by vendor.

Expected Output:.

```

> db.products.createIndex({ vendorId: 1 })
< vendorId_1

```

## 13. Text Search – Vendor Name Match

Enables full-text search on names.

Expected Output:

```

> db.vendors.createIndex({ name: "text" })
db.vendors.find({ $text: { $search: "Clara" } })
< {
  _id: 7,
  name: 'Clara Wells',
  phone: '416-555-6789',
  email: 'clara@example.com',
  startDate: 2023-09-17T00:00:00.000Z,
  isLocal: false
}

```

#### 14. Regex Query – Products with "knife"

Find products using pattern matching

Expected Output:

```
> db.products.find({ name: { $regex: /knife/i } })
< {
  _id: 103,
  vendorId: 3,
  name: 'Steel Hunting Knife',
  category: 'Knife',
  material: 'Carbon Steel',
  price: 45,
  inStock: 6
}
```

#### 15. Transaction Query