

MongoDB Advanced Queries Cheat Sheet

\$regex - Pattern Matching

```
db.users.find({ name: { $regex: 'roop', $options: 'i' } });
```

Use in: search bars, name filters

\$in / \$nin - Match from a List

```
db.products.find({ category: { $in: ['books', 'electronics'] } });
```

Use in: category filters

\$or / \$and - Combine Conditions

```
db.users.find({ $or: [{ isAdmin: true }, { age: { $gte: 18 } } ] });
```

Use in: admin filters

Date Range Queries

```
db.orders.find({ createdAt: { $gte: new Date('2024-01-01'), $lt: new Date('2024-12-31') } });
```

Use in: order filtering

\$lookup - Join Collections

```
db.orders.aggregate([ { $lookup: { from: 'users', localField: 'userId', foreignField: '_id', as: 'userDetails' } } ] );
```

Use in: show user with orders

\$group - Aggregation

```
db.orders.aggregate([ { $group: { _id: '$status', total: { $sum: 1 } } } ] );
```

Use in: dashboards

\$push / \$addToSet / \$pull - Array Operations

```
db.users.updateOne({ _id: userId }, { $addToSet: { favorites: 'book123' } });
```

Use in: wishlists

\$elemMatch - Array Matching

```
db.users.find({ cart: { $elemMatch: { productId: 'abc123', quantity: { $gte: 2 } } } });
```

Use in: cart systems

\$sort / \$limit / \$skip - Pagination

```
db.products.find().sort({ createdAt: -1 }).skip(10).limit(10);
```

Use in: product listing

\$text - Text Search

```
db.products.createIndex({ name: 'text', description: 'text' });
```

MongoDB Advanced Queries Cheat Sheet

```
db.products.find({ $text: { $search: 'laptop' } });
```

Use in: search with relevance

\$exists / \$type - Data Validation

```
db.users.find({ phone: { $exists: true, $type: 'string' } });
```

Use in: finding invalid entries

\$set / \$unset / \$rename - Field Operations

```
db.users.updateMany({}, { $rename: { 'fullname': 'name' } });
```

Use in: migrations