# MongoDB Compass cheat sheet

Docs for queries: <https://docs.mongodb.com/manual/reference/operator/query/>

Basic query structure: { field1: “arguments”, field2: “arguments” }

### Logical operators

|  |  |  |
| --- | --- | --- |
|  | **Operator** | **Example** |
| AND | $and | {$and:[{borough: "Queens"}, {cuisine: "American"}]}  Equivalent to:  {borough: “Queens”, cuisine: “American”} |
| OR | $or | {$or:[{borough: “Queens”}, {cuisine: “American”}]} |
| NOT | $ne | {cuisine: {$ne:”American”}} |
| $eq | {cuisine: {$eq:”American”}} |
| $nin | {cuisine: {$nin: [“Hamburgers”,”Bakery”]}} |
| $in | {cuisine: {$in: [“Hamburgers”,”Bakery”]}} |
| $exists | {cuisine:{$exists: false}} |

### Comparison query operators

|  |  |  |
| --- | --- | --- |
|  | **Operator** | **Example** |
| > | $gt | {runtime: {$gt: 60}} |
| ≥ | $gte | {runtime: {$gte: 60}} |
| < | $lt | {runtime: {$lt: 60}} |
| ≤ | $lte | {runtime: {$lte: 60}} |

String query

|  |  |  |
| --- | --- | --- |
|  | **Operator** | **Example** |
| Simple match | n/a | Gets all Document with “cuisine” field that exactly contains “Chinese”  {cuisine: “Chinese”}  Combine with $in or $nin to match multiple exact matches. E.g. to get all Document that have cuisine !== Chinese and American:  {cuisine: {$nin: [“Chinese”,”American”]}} |
| Regex | $regex: Pattern to match  $options: Flags. [Ref](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Regular_Expressions" \l "advanced_searching_with_flags).  [Ref](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Regular_Expressions) for $regex settings available | Gets all entries that have “cuisine” containing the letters “amer” e.g. american.  Flags used:  g: Global. Matches anywhere in ‘cuisine’ filed  i: Case insensitive  {cuisine: {regex:/amer/, settings:”gi”}}  Matches entries in ‘cuisine’ field that start with ‘american’  {cuisine: {regex:/^amer/}} |

### Querying objects and arrays

|  |  |  |
| --- | --- | --- |
|  | **Operation** | **Example** |
| Objects | Get Documents with specified value of field in object | {"tomatoes.viewer.rating": 3} |
| Arrays | Get value of array at specified index | {"genres.0":"Drama"} |
| Get all objects in an array field containing the specified value | {"genres":"Comedy"} |
| Get all Documents that have an object in an array field that has the specified values | {“scores.type”:”exam”, “scores.score”:{$gt: 90}} |

### Search options in MongoDB Compass

Graphical user interface, text, application, email

Description automatically generated

|  |  |
| --- | --- |
| **PROJECT** | Removes fields from the query result. Useful in large Collections containing Documents with many fields because it helps speed up the query because it reduces the size of all returned Documents.  Set to “0” to exclude a field, “1” to include just it |
| **SORT** | What this does depends on the target field’s value type. The value of each field for SORT can be either “1” or “-1"   * If the value is a Date, SORT arranges Documents by the earliest/latest date of the field * If the value is a String, SORT arranges Documents in by the field’s string value in alphabetical order * If the value is a number, SORT arranges by increasing/decreasing number |
| **LIMIT** | Specifies the maximum number of Documents to return |
| **SKIP** | Specifies how many Documents to skip from the search result starting from the first Document |
| **MAX TIME MS** | Specifies the maximum time in miliseconds to run the query |
| **COLLATION** | Allow setting [language-specific rules](https://docs.mongodb.com/upcoming/reference/collation/" \l "operations-that-support-collation) for string comparison, such as rules for lettercase and accent marks |