|  |  |  |  |
| --- | --- | --- | --- |
| **Operation** | **Syntax in MongoDb** | **Example** | **RDBMS Equivalent** |
| Equality | {<key>: {$eg: <value>}} | Db.post.find({‘by’: ‘user 1’}).pretty() | = |
| Less than | {<key>: {$lt: <value>}} | Db.post.find({‘likes’: {$lt: 50}}).pretty() | Where Likes < 50 |
| Less than Equals | {<key>: {$lte: <value>}} | Db.post.find({‘likes’: {‘$lte’: 50}}).pretty() | Where Likes <= 50 |
| Greater than | {<key>: {$gt: <value>}} | Db.post.find({‘likes’: ${‘gt’: 50}}).pretty() | Where Likes > 50 |
| Greater than Equals | {<key>: {$gte: <value>}} | Db.post.find({‘likes’: ${‘gte’: 50}}).pretty() | Where Likes >= 50 |
| Not Equals | {<key>: {$ne: <value>}} | Db.post.find({‘name’: {$ne: 50}).pretty() | Where Likes != 50 |
| Values in an Array | {<key>: {$in: [<val1>, <val2>, <val3>]}} | Db.post.find({‘name’: {$in: [‘val1’, ‘val2’]}}) | Where ‘name’ value in the array: [‘val1’, ‘val2’] |
| Values not in an Array | {<key>: {$nin: <val1>}} | Db.post.find({‘name’: {$nin: [‘val1’, ‘val2’]}}) | Where ‘name’ value is not in the array: [‘val1’, ‘val2’] |

Indexes: the method createIndex() also accepts the list of options

|  |  |  |
| --- | --- | --- |
| Parameter | Type | Description |
| background | Boolean | Builds the index in the background so that building an index does not block other database activities. Specify true to build in the background. The default value is false |
| unique | Boolean | Creates a unique index so that the collection will not accept insertion or documents where the index key or keys match an existing value in the index |
| name | String |  |
| sparse | Boolean | If true, the index only references documents with the specified field, these indexes use less space but behave differently in some situations. The default value is false |
| expireAfterSeconds | Integer |  |
| weights | String | Ranging from 1 to 99,999 and denotes the significance of the field relative to the other indexed fields in terms of the score |
| Language\_override | String | For a text index, specify the name of the field in the document that contains, the language to override the default language. The default value is language |

|  |  |  |
| --- | --- | --- |
| Expression | Description | Example |
| $sum | Sums up the defined value from all documents in the collection |  |
| $avg | Calculates the average of all the given values from all documents in the collection |  |
| $min | Gets the minimum of the corresponding values from all documents in the collection |  |
| $max | Gets the maximum… |  |
| $push | Inserts the value to an array in the resulting document |  |
| $addToSet | Inserts the value to an array in the resulting document but does not create duplicates |  |
| $first | Gets the first document from the source documents according to the grouping. Typically this makes only sense together with some previously applied ‘$sort’ stage |  |
| $last | Gets the last document… |  |