

NoSQL Cheatsheet

SQL

```
SELECT _id, name, address
FROM users
WHERE age > 18
```

NoSQL

```
db.users.find(
  { age: { $gt: 18 } }
  { name: 1, address: 1 } //displays name and address; _id
  displayed by default
```

```
SELECT * FROM inventory
WHERE status = "A"
AND qty < 30
```

```
db.inventory.find(
  { status: "A",
  qty: { $lt: 30 } } )
```

```
SELECT * FROM inventory
WHERE status = "A"
OR qty < 30
```

```
db.inventory.find(
  { $or: [ { status: "A" },
  { qty: { $lt: 30 } } ] } )
```

```
INSERT INTO users(name, age, status)
VALUES ("sue", 26, "A")
```

```
db.users.insert(
  {
    Name: "sue",
    Age: 26,
    Status: "A"
  }
)
```

//ADDITIONAL INSERT EXAMPLES BELOW

```
UPDATE users
SET status='A'
WHERE age > 18
```

```
db.users.update(
  { age: { $gt: 18 } }
  { $set: { status: "A" } }
```

// add , {multi: true} to the end to update multiple documents

```
)
```

```
DELETE FROM users
WHERE status = 'D'
```

```
db.users.remove(
  { status: "D" }
)
```

MongoDB also has deleteOne, deleteMany, insertOne, insertMany, updateOne, updateMany

Example - **INSERTS**: These examples contain **embedded documents** (documents/records inside documents/records) in the example of size

```
db.inventory.insertOne(
  { item: "canvas", qty: 100, tags: ["cotton"], size: { h: 28, w: 35.5, uom: "cm" } }
)
```

```
db.inventory.insertMany([
  { item: "journal", qty: 25, tags: ["blank", "red"], size: { h: 14, w: 21, uom: "cm" } },
  { item: "mat", qty: 85, tags: ["gray"], size: { h: 27.9, w: 35.5, uom: "cm" } },
  { item: "mousepad", qty: 25, tags: ["gel", "blue"], size: { h: 19, w: 22.85, uom: "cm" } }
])
```

Primary key is `_id`. If the document does not specify an `_id` field, MongoDB adds the `_id` field with an `ObjectId` value to the new document