

# Mapping SQL to MongoDB

## Converting to MongoDB Terms

MYSQL EXECUTABLE	ORACLE EXECUTABLE	MONGODB EXECUTABLE
mysqld	oracle	mongod
mysql	sqlplus	mongo

SQL TERM	MONGODB TERM
database	database
table	collection
index	index
row	document
column	field
joining	embedding & linking

Queries and other operations in MongoDB are represented as documents passed to **find** and other methods. Below are examples of SQL statements and the analogous statements in MongoDB JavaScript shell syntax.

SQL	MONGODB
CREATE TABLE users (name VARCHAR(128), age NUMBER)	db.createCollection("users")
INSERT INTO users VALUES ('Bob', 32)	db.users.insert({name: "Bob", age: 32})
SELECT * FROM users	db.users.find()
SELECT name, age FROM users	db.users.find({}, {name: 1, age: 1, _id: 0})
SELECT name, age FROM users WHERE age = 33	db.users.find({age: 33}, {name: 1, age: 1, _id: 0})
SELECT * FROM users WHERE age > 33	db.users.find({age: {\$gt: 33}})
SELECT * FROM users WHERE age <= 33	db.users.find({age: {\$lte: 33}})

SQL	MONGODB
SELECT * FROM users WHERE age > 33 AND age < 40	db.users.find({age: {\$gt: 33, \$lt: 40}})
SELECT * FROM users WHERE age = 32 AND name = 'Bob'	db.users.find({age: 32, name: "Bob"})
SELECT * FROM users WHERE age = 33 OR name = 'Bob'	db.users.find({\$or:[{age:33}, {name: "Bob"}]})
SELECT * FROM users WHERE age = 33 ORDER BY name ASC	db.users.find({age: 33}).sort({name: 1})
SELECT * FROM users ORDER BY name DESC	db.users.find().sort({name: -1})
SELECT * FROM users WHERE name LIKE 'Joe%'	db.users.find({name: /Joe/})
SELECT * FROM users WHERE name LIKE 'Joe%'	db.users.find({name: /^Joe/})
SELECT * FROM users LIMIT 10 SKIP 20	db.users.find().skip(20).limit(10)
SELECT * FROM users LIMIT 1	db.users.findOne()
SELECT DISTINCT name FROM users	db.users.distinct("name")
SELECT COUNT(*) FROM users	db.users.count()
SELECT COUNT(*) FROM users WHERE AGE > 30	db.users.find({age: {\$gt: 30}}).count()
SELECT COUNT(AGE) FROM users	db.users.find({age: {\$exists: true}}).count()
UPDATE users SET age = 33 WHERE name = 'Bob'	db.users.update({name: "Bob"}, {\$set: {age: 33}}, {multi: true})
UPDATE users SET age = age + 2 WHERE name = 'Bob'	db.users.update({name: "Bob"}, {\$inc: {age: 2}}, {multi: true})
DELETE FROM users WHERE name = 'Bob'	db.users.remove({name: "Bob"})
CREATE INDEX ON users (name ASC)	db.users.ensureIndex({name: 1})
CREATE INDEX ON users (name ASC, age DESC)	db.users.ensureIndex({name: 1, age: -1})
EXPLAIN SELECT * FROM users WHERE age = 32	db.users.find({age: 32}).explain()