SQL to MongoDB Mapping

SQL	MongoDB
CREATE TABLE lju (id MEDIUMINT NOT NULL AUTO_INCREMENT, user_id Varchar(20),age Number, status char(1), PRIMARY KEY (id))	**db.createCollection ("lju”)** creates collection no need to define schema
ALTER TABLE lju ADD join_date DATETIME	db.lju.updateMany({ }, { $set: { join_date: new Date() } }) **Note:** **new Date().toLocaleDateString()-The toLocaleDateString() method returns the date (not the time) of a date object as a string, using locale conventions.**
ALTER TABLE lju DROP COLUMN join_date	db.lju.updateMany({ }, { $unset: { "join_date": "" } })
DROP TABLE lju	db.lju.drop()

INSERT INTO lju (name, age, status)
VALUES (“Test”, 45, "A")
	db.lju.insertOne(
	{ name: “Test”, age: 18, status: "A" })

SELECT *FROM lju	db.lju.find()
SELECT id, name FROM lju	db.lju.find({ }, { name:1 })
SELECT name, status FROM lju	db.lju.find({ }, { name:1,status:1,_id: 0 })
SELECT * FROM lju WHERE status ="A"	db.lju.find({ status: "A" })

SQL to MongoDB Mapping

SELECT name FROM lju WHERE status = "A"	db.lju.find({ status: "A" }, { name: 1,_id: 0 })
SELECT * FROM lju WHERE status != "A"	db.lju.find({ status: { $ne: "A" } })

SELECT *
FROM lju
WHERE status = "A"
AND age = 50
	db.lju.find({status:”A”,age:50})
	or
	db.lju.find(
	{ $and: [{ status: "A" } , { age: 50 }] }
	)

SELECT * FROM lju
WHERE status = "A"
OR age = 30
	db.lju.find(
	{ $or: [{ status: "A" } , { age: 30 }] }
	)

SELECT * FROM lju
WHERE age > 25
	db.lju.find(
	{ age: { $gt: 25 } }
	)

SELECT * FROM lju
WHERE age <= 25
	db.lju.find(
	{ age: { $lte: 25 } }
	)

SELECT * FROM lju
WHERE age > 25
AND age<= 50
	db.lju.find(
	{ age: { $gt: 25, $lte: 50 } }
	)

SELECT *
FROM lju
WHERE name like "%abc%"
	db.lju.find({ name: {$regex: /abc/ }})
	or
	db.lju.find({ name: { $regex: /abc/ } })

SELECT *
FROM lju
WHERE name like "abc%"
	db.lju.find({ name: /^bc/ })
	or
	db.lju.find({ name: { $regex: /^abc/ } })

SELECT * from lju where name=”abc” and age=20	db.lju.find({name:"abc",age:20})

SELECT *
FROM lju
WHERE status = "A"
ORDER BY age ASC
	db.lju.find({ status: "A" }).sort({ age: 1 })

SELECT *
FROM lju
WHERE status = "A"
ORDER BY age DESC
	db.lju.find({ status: "A" }). sort({ age: -1 })

SQL to MongoDB Mapping

SELECT COUNT(*) FROM lju	db.lju. count() or db. lju. find(). count()

SELECT COUNT(*)
FROM lju
WHERE age > 30
	db.lju.count({ age: { $gt: 30 } })
	or
	db.lju.find({ age: { $gt: 30 } }).count()

SELECT *
FROM lju
LIMIT 1
	db.lju.findOne()or
	db.lju.find().limit(1)

SELECT * FROM lju LIMIT 2 OFFSET 3;	db.lju.find().limit(2).skip(3)
EXPLAIN SELECT * FROM lju WHERE status = "A"	db.lju. find({ status: "A" }).explain()

UPDATE lju SET status = "C"
WHERE age > 25
	db.lju.updateMany({ age: { $gt: 25 } }, { $set: { status: "C" } })

UPDATE lju SET age = age + 3
WHERE status = "A"
	db.lju.updateMany({ status: "A" } , { $inc: { age: 3 }

DELETE FROM lju WHERE status = "D"	db.lju.deleteMany({ status: "D" })
DELETE FROM lju	db.lju.deleteMany({ })

SELECT *
FROM lju
WHERE name NOT IN ('aaa','abc','bbb');
	db.lju.find({ name: { $nin: [‘aaa’,’abc’,’bbb’] } })

SELECT name
FROM lju
WHERE age IN (20,23,33);
	db.lju.find({ age: { $in: [20, 23, 33] } },{name:1,_id:0})