

# SQL vs MongoDB

- SQL:
- `SELECT *`
- `FROM people`
  
- MongoDB:
- `db.people.find({})`

# SQL vs MongoDB

- SQL:
- `SELECT id,`
- `user_id,`
- `status`
- `FROM people`
  
- MongoDB:
- `db.people.find({}, { _id: 1, user_id: 1, status: 1  
    })`

# SQL vs MongoDB

- SQL:
- `SELECT user_id, status`
- `FROM people`
- MongoDB:
- `db.people.find({}, { user_id: 1, status: 1 })`

# SQL vs MongoDB

- SQL:
- `SELECT *`
- `FROM people`
- `WHERE status = "A"`
- MongoDB:
- `db.people.find({ status: "A" })`

# SQL vs MongoDB

- SQL:
- `SELECT user_id, status`
- `FROM people`
- `WHERE status = "A"`
- MongoDB:
- `db.people.find({ status: "A" }, { user_id: 1, status: 1 })`

# SQL vs MongoDB

- SQL:
- `SELECT *`
- `FROM people`
- `WHERE status != "A"`
- MongoDB:
- `db.people.find({ status: { $ne: "A" } })`

# SQL vs MongoDB

- SQL:
- `SELECT *`
- `FROM people`
- `WHERE status = "A"`
- `AND age = 50`
- MongoDB:
- `db.people.find({ status: "A", age: 50 })`

# SQL vs MongoDB

- SQL:
- `SELECT *`
- `FROM people`
- `WHERE status = "A"`
- `OR age = 50`
- MongoDB:
- `db.people.find({ $or: [{ status: "A" }, { age: 50 }] })`



# SQL vs MongoDB

- SQL:
- `SELECT *`
- `FROM people`
- `WHERE age > 25`
- MongoDB:
- `db.people.find({ age: { $gt: 25 } })`

# SQL vs MongoDB

- SQL:
- `SELECT *`
- `FROM people`
- `WHERE age < 25`
  
- MongoDB:
- `db.people.find({ age: { $lt: 25 } })`

# SQL vs MongoDB

- SQL:
- `SELECT *`
- `FROM people`
- `WHERE age > 25`
- `AND age <= 50`
- MongoDB:
- `db.people.find({ age: { $gt: 25, $lte: 50 } })`

# SQL vs MongoDB

- SQL:
- `SELECT *`
- `FROM people`
- `WHERE user_id like "%bc%"`
- MongoDB:
- `db.people.find({ user_id: /bc/ })`

# SQL vs MongoDB

- SQL:
- `SELECT *`
- `FROM people`
- `WHERE status = "A"`
- `ORDER BY user_id ASC`
- MongoDB:
- `db.people.find({ status: "A" }).sort({ user_id: 1 })`

# SQL vs MongoDB

- SQL:
- `SELECT *`
- `FROM people`
- `WHERE status = "A"`
- `ORDER BY user_id DESC`
- MongoDB:
- `db.people.find({ status: "A" }).sort({ user_id: -1 })`

# SQL vs MongoDB

- SQL:
- `SELECT COUNT(*)`
- `FROM people`
  
- MongoDB:
- `db.people.countDocuments({})`

# SQL vs MongoDB

- SQL:
- `SELECT COUNT(user_id)`
- `FROM people`
  
- MongoDB:
- `db.people.countDocuments({ user_id: { $exists: true } })`



# SQL vs MongoDB

- SQL:
- `SELECT COUNT(*)`
- `FROM people`
- `WHERE age > 30`
- MongoDB:
- `db.people.countDocuments({ age: { $gt: 30 } })`

# SQL vs MongoDB

- SQL:
- `SELECT DISTINCT(status)`
- `FROM people`
  
- MongoDB:
- `db.people.distinct("status")`

# SQL vs MongoDB

- SQL:
- `SELECT *`
- `FROM people`
- `LIMIT 1`
  
- MongoDB:
- `db.people.findOne({})`

# SQL vs MongoDB

- SQL:
- `SELECT *`
- `FROM people`
- `LIMIT 5`
- `SKIP 10`
- MongoDB:
- `db.people.find({}).skip(10).limit(5)`

# SQL vs MongoDB

- SQL:
- CREATE TABLE people (- id MEDIUMINT NOT NULL
- AUTO\_INCREMENT,
- user\_id Varchar(30),
- age Number,
- status char(1),
- PRIMARY KEY (id)
- )

# SQL vs MongoDB

- SQL:
- ALTER TABLE people
- ADD join\_date DATETIME
- MongoDB:
- `db.people.updateMany({}, { $set: { join_date: null } })`

# SQL vs MongoDB

- SQL:
- `DROP TABLE people`
- MongoDB:
- `db.people.drop()`

# SQL vs MongoDB

- SQL:
- `INSERT INTO people(user_id, age, status)`
- `VALUES ("bcd001", 45, "A")`
  
- MongoDB:
- `db.people.insertOne({ user_id: "bcd001", age: 45, status: "A" })`



# SQL vs MongoDB

- SQL:
- UPDATE people
- SET status = "C"
- WHERE age > 25
- MongoDB:
- `db.people.updateMany({ age: { $gt: 25 } }, { $set: { status: "C" } })`

# SQL vs MongoDB

- SQL:
- UPDATE people
- SET age = age + 3
- WHERE status = "A"
  
- MongoDB:
- `db.people.updateMany({ status: "A" }, { $inc: { age: 3 } })`

# SQL vs MongoDB

- SQL:
- `DELETE FROM people`
- `WHERE status = "D"`
- MongoDB:
- `db.people.deleteMany({ status: "D" })`

# SQL vs MongoDB

- SQL:
- `DELETE FROM people`
- MongoDB:
- `db.people.deleteMany({})`