1: SELECT students.\* FROM students WHERE students.grade > 90

2: SELECT name, age FROM users ORDER BY age DESC

3: UPDATE products SET price = price + (price \* 0.1) WHERE category = 'Electronics'

4: INSERT INTO employees (name, position, salary) VALUES ('John Smith', 'Manager', 75000)

5: DELETE FROM orders WHERE order\_date < '2023-01-01'

6: SELECT c.name, COUNT(o.order\_id) FROM customers c INNER JOIN orders o ON c.customer\_id = o.customer\_id GROUP BY c.name

7: CREATE TABLE projects (project\_id INTEGER PRIMARY KEY, name VARCHAR(100), start\_date DATE)

8: ALTER TABLE users ADD COLUMN last\_login TIMESTAMP

9: SELECT AVG(e.salary) FROM employees e WHERE e.department = 'Marketing'

10: CREATE INDEX idx\_lastname ON customers(last\_name)

11: db.collection.find({ status: "active" })

12: db.users.updateOne({ \_id: ObjectId("507f191e810c19729de860ea") }, { $set: { status: "inactive" }})

13: db.products.aggregate([{ $match: { category: "Electronics" }}, { $group: { \_id: "$brand", count: { $sum: 1 }}}])

14: db.orders.deleteMany({ orderDate: { $lt: new Date("2023-01-01") }})

15: db.users.createIndex({ email: 1 }, { unique: true })

16: MATCH (u:User)-[:ORDERED]->(p:Product) WHERE p.category = 'Books' RETURN u.name, COUNT(p) as order\_count

17: SELECT name, department FROM employees WHERE salary >= 50000 AND salary <= 70000

18: SELECT p.product\_name, c.category\_name FROM products p INNER JOIN categories c ON p.category\_id = c.category\_id

19: db.inventory.updateMany({}, { $inc: { quantity: -1 } })

20: WITH orders AS (SELECT customer\_id, COUNT(\*) as order\_count FROM orders GROUP BY customer\_id) SELECT c.name, o.order\_count FROM customers c JOIN orders o ON c.customer\_id = o.customer\_id WHERE o.order\_count > 5