|  |  |
| --- | --- |
| SQL SELECT Statement | MongoDB find() Statement |
| SELECT \* FROM Employee | db.Employee.find() |
| SELECT id, Employeeid, EmployeeName FROM Employee | db.Employee.find( { }, { Employeeid: 1, EmployeeName: 1 } ) |
| SELECT Employeeid, EmployeeName FROM Employee | db.Employee.find( { }, { Employeeid: 1, EmployeeName: 1, \_id: 0 } ) |
| SELECT \* FROM Employee WHERE EmployeeName = "B" | db.Employee.find( { EmployeeName: "A" } ) |
| SELECT Employeeid, EmployeeName FROM Employee WHERE EmployeeName = "A" | db.Employee.find( { EmployeeName: "A" }, { Employeeid: 1, EmployeeName: 1, \_id: 0 } ) |
| SELECT \* FROM Employee WHERE EmployeeName != "A" | db.Employee.find( { EmployeeName: { $ne: "A" } } ) |
| SELECT \* FROM Employee WHERE EmployeeName = "A" AND Employeeid = 50 | db.Employee.find( { EmployeeName: "A", Employeeid: 50 } ) |
| SELECT \* FROM Employee WHERE EmployeeName = "A" OR Employeeid = 50 | db.Employee.find( { $or: [ { EmployeeName: "A" } , { Employeeid: 50 } ] } ) |
| SELECT \* FROM Employee WHERE Employeeid > 25 | db.Employee.find( { Employeeid: { $gt: 25 } } ) |
| SELECT \* FROM Employee WHERE Employeeid < 25 | Db.Employee.find( { Employeeid: { $lt: 25 } } ) |
| SELECT \* FROM Employee WHERE Employeeid > 25 AND Employeeid <= 50 | db.Employee.find( { Employeeid: { $gt: 25, $lte: 50 } } ) |
| SELECT \* FROM Employee WHERE Employeeid like "%bc%" | db.Employee.find( { Employeeid: /bc/ } ) -or- db.Employee.find( { Employeeid: { $regex: /bc/ } } ) |
| SELECT \* FROM Employee WHERE Employeeid like "bc%" | db.Employee.find( { Employeeid: /^bc/ } ) -or- db.Employee.find( { Employeeid: { $regex: /^bc/ } } ) |
| SELECT \* FROM Employee WHERE EmployeeName = "A" ORDER BY Employeeid ASC | db. Employee. find( { EmployeeName: "A" } ). sort( { Employeeid: 1 } ) |
| SELECT \* FROM Employee WHERE EmployeeName = "A" ORDER BY Employeeid ASC | db. Employee. find( { EmployeeName: "A" } ). sort( { Employeeid: 1 } ) |
| SELECT \* FROM Employee WHERE EmployeeName = "A" ORDER BY Employeeid ASC | db. Employee. find( { EmployeeName: "A" } ). sort( { Employeeid: 1 } ) |
| SELECT \* FROM Employee WHERE EmployeeName = "A" ORDER BY Employeeid DESC | db. Employee. find( { EmployeeName: "A" } ). sort( { Employeeid: -1 } ) |
| SELECT \* FROM Employee WHERE EmployeeName = "A" ORDER BY Employeeid DESC | db. Employee. find( { EmployeeName: "A" } ). sort( { Employeeid: -1 } ) |
| SELECT COUNT(\*) FROM Employee | db. Employee. count() or db. Employee. find(). count() |
| SELECT COUNT(Employeeid) FROM Employee | db. Employee.count( { Employeeid: { $exists: true } } ) or db. Employee.find( { Employeeid: { $exists: true } } ).count() |
| SELECT COUNT(\*) FROM Employee WHERE Employeeid > 30 | db. Employee.count( { Employeeid: { $gt: 30 } } ) or db. Employee.find( { Employeeid: { $gt: 30 } } ).count() |
| SELECT DISTINCT(EmployeeName) FROM Employee | db. Employee.aggregate( [ { $group : { \_id : "$EmployeeName" } } ] ) or, for distinct value sets that do not exceed the BSON size limit db. Employee.distinct( "EmployeeName" ) |
| SELECT \* FROM Employee LIMIT 1 | db. Employee.findOne() or db. Employee.find(). limit(1) |
| SELECT \* FROM Employee LIMIT 5 SKIP 10 | db. Employee.find(). limit(5). skip(10) |
| EXPLAIN SELECT \* FROM Employee WHERE EmployeeName = "A" | db. Employee. find( { EmployeeName: "A" } ). explain() |