Q1. Point out the wrong statement:

1. **Indexes cannot enforce uniqueness in collection**
2. CRUD stands for create, read, update, and delete
3. CRUD application is the most simplest application
4. All of the mentioned

Q2. \_\_\_\_\_\_ returns true if the first value is less than the second.

1. $lte
2. **$lt**
3. $setEquals
4. None of the mentioned

Q3. $ne returns true if the values are not \_\_\_\_\_\_\_ .

1. **equivalent**
2. null
3. 0
4. None of the mentioned

Q4. In $min operator, if the key is having null, than what value $min will returns.

1. 1
2. 0
3. **null**
4. None of the mentioned

Q5. In $max operator, if the key is having null, than what value $max will returns.

1. 1
2. 0
3. **null**
4. None of the mentioned

Q6. In $sum operator, if the key is having null, than what value $sum will returns.

1. 1
2. **0**
3. null
4. None of the mentioned

Q7. Which of the following is equivalent to this:

select first\_name, salary from employees where designation=”Manager”;

Assume that there are three columns first\_name, salary, date\_of\_join.

1. **db.employees.find({"designation":"Manager"},{"date\_of\_join" : 0})**
2. db.employees.find({"designation:Manager"},{"date\_of\_join" : 0})
3. db.employees.find({"designation:Manager"},{"date\_of\_join" : 1})
4. None of the above.

Q8. Which of the following MongoDB conditional operator is not a valid operator?

1. $lte
2. $gt
3. **$gtr**
4. $lt

Q9. What is equivalent MongoDB query for the following SQL statement.

SELECT \* FROM emp WHERE deptno = 30 ORDER BY deptno DESC;

1. **db.emp.find( {deptno: 30 } ).sort({ deptno: -1 } )**
2. db.emp.find( {deptno= 30 } ).sort({ deptno: 1 } )
3. db.emp.find( {deptno: 30 } ).orderby({ deptno: -1 } )
4. an error in statement.

Q10. What is equivalent MongoDB query for the following SQL statement.

INSERT INTO emp values("abc123", 55, "A");

1. db.people.Insert( { user\_id: "abc123", age: 55, status: "A" } )
2. db.people.insertone( { user\_id: "abc123", age: 55, status: "A" } )
3. **db.people.insertOne( { user\_id: "abc123", age: 55, status: "A" } )**
4. an error in statement.

Q11.

Q12.

Q13.

Q14.

Q15.

Q16.

Q17.

Q18.

Q19. What is equivalent MongoDB query for the following SQL statement.

INSERT INTO emp values("abc123", 55, "A");

1. **db.people.insert( { user\_id: "abc123", age: 55, status: "A" } )**
2. db.people.insertone( { user\_id: "abc123", age: 55, status: "A" } )
3. db.people.insertOnes( { user\_id: "abc123", age: 55, status: "A" } )
4. an error in statement.

Q20.

Q21.

Q22.

Q23.

Q24.

Q25.

Q26.

Q27. What is equivalent MongoDB query for the following SQL statement.

INSERT INTO emp values("abc123", 55, "A");

1. db.people.insertmany( { user\_id: "abc123", age: 55, status: "A" } )
2. db.people.insertMany( { user\_id: "abc123", age: 55, status: "A" } )
3. **db.people.insertMany( [ { user\_id: "abc123", age: 55, status: "A" } ] )**
4. an error in statement.

Q28.

Q29.

Q30.

Q31.

Q32.

Q33.

Q34.

Q35.

Q36.

Q37.

Q38.

Q39.

Q40.

Q41.

Q42.

Q43.

Q44.

Q45.

Q46.

Q47.

Q48.

Q49.

Q50.