**Assignment 09july**

**1)How many documents are there in the collection?**

🡪db.user.countDocuments()

21

**2)Retrieve all documents where the role is "admin".**

🡪db.user.find({role:"admin"})

**3)Find all documents where the role is "customer" and the username contains "Karamchandani".**

🡪db.user.find({"$and":[{role:"customer"},{username:{"$regex":"Karamchandani","$options":"i"}]})

**4)Retrieve the document with the username "Sumanth Rao".**

**🡪**db.user.find({username:"Sumanth Rao"})

**5)Find all documents where the user's first name starts with "R".**

🡪db.user.find({username:/^R/})

**6)Retrieve all documents where the user has the skill "Python".**

🡪db.user.find({skills:"Python"})

**7)Find all documents where the user has the skill "Java" and "Pythosn".**

🡪 db.user.find({"$and":[{skills:"Python"},{skills:"Java"}]})

**8)Retrieve all documents where the user's first name is "Aryan" or the last name is "Gupta".**

🡪db.user.find({"$or":[{firstName:"Aryan"},{lastName:"Gupta"}]})

**9)Find all documents where the user has the skill "Java" but not "Python".**

🡪db.users.find({"$and":[{skills:"Java"},{skills:{"$ne":"Python"}}]})

**10)Retrieve all documents where the user's last name is "Sharma" and the role is "customer".**

🡪db.user.find({"$and":[{lastName:"Sharma"},{role:"customer"}]})

**11)Find all documents where the user has exactly three skills.**

🡪 db.user.find({skills:{"$size":3}})

**12)Retrieve all documents where the user's first name starts with "S" and the role is "admin".**

🡪db.user.find({"$and":[{role:"admin"},{FirstName:/^S/}]})

**13)Find all documents where the user has the skill "Java" and the role is not "admin".**

**🡪**db.user.find({"$and":[{role:{"$ne":"admin"}},{skills:"Java"}]})

**14)Retrieve all documents where the user's role is not specified.**

🡪 db.user.find({role:{"$exists": false }})

**15)Find all documents where the user has at least four skills.**

🡪 db.user.find({"$expr": {"$gte":[{"$size":"$skills"},4]}})

**16)Retrieve all documents where the user's role is "customer" and the user has no specified skills.**

**🡪**db.user.find({role:"customer",skills:{"$exists":false}})

**17)Find all documents where the user's email contains "regexsoftware.com".**

🡪db.user.find({useremail: {"$regex":"regexsoftware\\.com","$options":"i”}})

**18)Retrieve all documents where the user's email does not contain "gmail.com".**

🡪db.user.find({useremail:{"$not":{"$regex":"gmail\\.com","$options":"i"}}})

**19)Find all documents where the user's last name is not "Singh".**

🡪db.user.find({lastName:{"$ne":"Singh"}})

**20)Retrieve all documents where the user's role is "admin" and the user has the skill "Java".**

🡪db.user.find({"$and":[{role:"admin"},{skills:"Java"}]})

**21)Find all documents where the user's first name is "Ananya" and the user's role is "customer".**

🡪db.user.find({"$and":[{firstName:"Ananya"},{role:"customer"}]})

**22)Retrieve all documents where the user's role is "admin" and the user has at least one skill.**

🡪 db.user.find({role:"admin",skills:{"$exists":true,"$ne":[]})

**23)Find all documents where the user's role is "admin" and the user's first name starts with "R".**

🡪db.user.find({"$and":[{role:"admin"},{FirstName:/^R/}]})

**24)Retrieve all documents where the user's role is "customer" and the user's last name is "Patel".**

**🡪**db.user.find({"$and":[{role:"customer"},{lastName:"Patel"}]})

**25)Find all documents where the user's role is "admin" and the user has the skill "Python".**

**🡪**db.user.find({"$and":[{role:"admin"},{skills:"Python"}]})

**26)Retrieve all documents where the user's role is "customer" and the user's last name is not "Gupta".**

**🡪**db.user.find({"$and":[{role:"customer"},{lastName:{"$ne":"Patel"}}]})

**27)Find all documents where the user's role is "admin" and the user's email ends with "@regexsoftware.com".**

**🡪**db.user.find{“$and”:[{role:”admin”},{

**28)Retrieve all documents where the user's role is "customer" and the user has the skill "React".**

🡪db.user.find({"$and":[{role:"customer"},{skills:"React"}]})

**29)`Find all documents where the user's role is "customer" and the user has the skill "SQL".**

🡪 db.user.find({"$and":[{role:"customer"},{skills:"SQL"}]})

**30)Retrieve all documents where the user's role is "admin" and the user has at least three skills.**

🡪db.user.find({"$and":[{role:"admin"},{skills:{"$sizeal":3}}]})

**31)Find all documents where the user's role is "customer" and the user has the skills "Java" and "C++".**

🡪db.user.find({"$and":[{role:"customer"},{skills:"Java"},{skills"C++"}]})

**32)Retrieve all documents where the user's role is "admin" and the user's email does not contain "gmail.com".**

**🡪** db.user.find({role:"admin",useremail:{"$not":{"$regex":"gmail\\.com","$options":"i"}}})

**33)Find all documents where the user's role is "customer" and the user's last name starts with "M".**

🡪db.user.find({"$and":[{role:"customer"},{lastName:/^M/}]})

**34)Retrieve all documents where the user's role is "admin" and the user's first name is not "Rahul".**

🡪db.user.find({"$and":[{role:"admin"},{firstName:{"$ne":"Rahul"}}]})

**35)Find all documents where the user's role is "customer" and the user's email does not contain "gmail.com".**

**🡪** db.user.find({role:"customer",useremail:{"$not":{"$regex":"gmail\\.com","$options":"i"}}})

**36)Retrieve all documents where the user's role is "admin" and the user has the skill "Java" but not "JavaScript".**

🡪 db.user.find({"$and":[{role:"admin"},{skills:{"$ne":"JavaScript"}},{skills:"Java"}]})

**37)Find all documents where the user's role is "customer" and the user's first name is not "Ananya".**

🡪db.user.find({"$and":[{firstName:{"$ne":"Ananya"}},{role:"customer"}]})

**38)Retrieve all documents where the user's role is "admin" and the user's email contains "regexsoftware.com".**

**🡪** db.user.find({role: "admin",useremail:{"$regex":"regexsoftware\\.com","$options": "i" }})

**39)Find all documents where the user's role is "customer" and the user has the skill "HTML" but not "CSS".**

🡪db.user.find({"$and":[{role:"customer"},{skills:{"$ne":"CSS"}},{skills:"HTML"}]})

**40)Retrieve all documents where the user's role is "admin" and the user has the skill "Java" and "JavaScript".**

🡪 db.user.find({"$and":[{role:"admin"},{skills:"JavaScript"},{skills:"Java"}]})

**41)Find the average age of users.**

🡪 [{$group: {\_id: "$user",averageage:{$avg:"$age"}}}]

**42)Find all female users from Jaipur who know Python.**

🡪db.user.find({"$and":[{gender:"Female"},{city:"Jaipur"},{skills:"Python"}]})

**43)Find all male users in Delhi who are above age 26**

🡪 db.user.find({"$and":[{gender:"Male"},{city:"Delhi"},{age:{"$gt":26}}]})

**44)Find all users who are from either New Delhi or Mumbai.**

🡪db.user.find({"$or":[{city:"Delhi"},{city:"Mumbai"}]})

**45)Find all users who have the same last name.**

**🡪** [{$group: {

\_id: "$lastName",

usercount:{$sum:1},

users:{$push:"$$ROOT"}}}]

**46)Find the count of users in each city.**

🡪 [

{ $group: {

\_id:"$city",

countofusers:{$sum:1}

}}

]

**47)How many users are from New Delhi?**

🡪db.user.find({city:"New Delhi"})

**48)How many Female users are from Jaipur?**

🡪db.user.find({"$and":[{gender:"Female"},{city:"Jaipur"}]})

**49)How many Male users are there in Delhi who have skills Java?**

🡪 db.user.find({"$and":[{gender:"Male"},{city:"Delhi"},{skills:"Java"}]})

**50)How many users are from Hyderabad having age greater than 24 and have skill Python,C++**

🡪 db.user.find({"$and":[{gender:"Female"},{age:{"$gt":24}},{skills:"Python"},{city:"Hyderabad"}{skills:"C++"}]})

**51)How many female users are from Goa having age greater than 24 and have skill Python,C++**

🡪db.user.find({"$and":[{gender:"Female"},{age:{"$gt":24}},{skills:"Python"},{skills:"C++"},{city:"Goa"}]})

1. Group users by role and calculate the average salary for each role.

🡪 [{

$group: {

\_id:"$role",

averagSalary:{$avg:"$salary"}

}

}

]

2. Group users by city and find the total number of users in each city.

🡪[{

$group: {

\_id: '$city',

users:{$sum:1}

}

}]  
3. Group users by gender and determine the average age for each gender.

[{

$group: {

\_id: '$gender',

averageage:{$avg:"$age"}

}

}]  
4. Group users by role and city, and calculate the total salary for each subgroup

[{

$group: {

\_id: {city:"$city",gender:"$gender"},

"totalSalary": {$sum:"$salary"}

}},

]

5. Group users by gender and city, and find the maximum salary in each subgroup.

[{

$group:{

\_id:{"gender":"$gender","city":"$city"},

maxSalary:{"$max":"$salary"}

}

}

]  
6. Group users by skills and count how many users have each skill.

[

{$unwind:"$skills"

},{$group: {

\_id:"$skills",

countskills:{$sum:1}

}

}

]  
7. Group users by role and find the minimum age for each role.

[

{$group: {

\_id:"$role",

minage:{$min:"$age"}

}

}

]  
8. Group users by city and gender, and calculate the average salary for each subgroup.

[

{$group: {

\_id:{"gender":"$gender","city":"$city"},

averageSalary:{$avg:"$salary"}

}

}

]  
9. Group users by role and determine the total number of unique cities for each role.

[

{

$group: {

\_id: "$role",

cities: { $addToSet: "$city" }

}

},

{

$project: {

\_id: 0,

role: "$\_id",

numberOfCities: { $size: "$cities" }

}

}

]  
10. Group users by age and count the number of users in each age group.

[

{

$group: {

\_id: "$age",

count: { "$sum": 1 }

}

}

]

11. Group users by city and calculate the average number of skills per user in each city.

[{

$unwind: "$skills"

},

{

$group: {

\_id: "$city",

avgskills:{$avg:{$size:"$skills"}}

}

},

{

$project: {

city:"$\_id.city",

avgskills:1

}

}

]  
12. Group users by role and find the highest salary in each role.

[

{

$group: {

\_id: "$role",

highestSalary:{$max:"$salary"}

}

}

]  
13. Group users by gender and calculate the total number of users for each gender.

[

{

$group: {

\_id: "$gender",

count:{$sum:1}

}

}

]  
14. Group users by city and determine the average age for users in each city. [

{

$group: {

\_id: "$city",

averageAge:{$avg:"$age"}

}

}

]

15. Group users by role and gender, and calculate the average salary for each subgroup.

[

{$group: {

\_id:{"gender":"$gender","role":"$role"},

averageSalary:{$avg:"$salary"}

}

}

]  
16. Group users by skills and determine the total number of users for each skill.

[{$unwind:"$skills"},

{$group: {

\_id:"$skills",

users:{$sum:1}

}

}

]  
17. Group users by age and find the maximum salary for each age group.

[

{$group: {

\_id:"$age",

averageSalary:{$avg:"$salary"}

}

}

]  
18. Group users by role and calculate the total number of users with more than three skills.

[

{

$match: {

$expr: { $gt: [{ $size: "$skills" }, 3] }

}

},

{

$project: {

\_id: 1,

role: 1,

}

},

{

$group: {

\_id: "$role",

count: { $sum: 1 },

}

}

]  
19. Group users by gender and city, and find the average number of skills per user in each subgroup.

[

{$group: {

\_id:{"gender":"$gender","city":"$city"},

averageskills:{$avg:{$size:"$skills"}}

}

}

]  
20. Group users by role and determine the highest and lowest salaries for each role.

[

{$group: {

\_id:"$role",

highestSalary:{$max:"$salary"},

lowestSalary:{$min:"$salary"}

}

}

]

21. Group users by role and city, and calculate the total number of users with more than one skill.  
22. Group users by gender and city, and determine the total number of users with a salary less than 500,000 in each subgroup.  
23. Group users by age and find the average salary of users who have "customer" role in each age group.

[{

$match:

{

role:"customer"

}

},

{

$group:{

\_id:"$age",

avgsalary:{$avg:"$salary"}

}}

]

24. Group users by role and determine the average salary for users with "Kotlin" as a skill.

[{

$match:

{

skills:"Kotlin"

}

},

{

$group:{

\_id:"$role",

avgsalary:{$avg:"$salary"}

}}

]  
25. Group users by city and calculate the total number of users with "admin" role in each city.

[{

$match:

{

role:"admin"

}

},

{

$group:{

\_id:"$city",

count:{$sum:1}

}},

{

$project:{

role:"admin",

count:1

}

}

]  
26. Group users by gender and determine the total number of users with "Java" and "Python" both as skills.

[{

$match:

{

skills:["Java","Python"]

}

},

{

$group:{

\_id:"$gender",

count:{$sum:1}

}},

{

$project:{

skills:["Java","Python"],

count:1

}

}

]  
27. Group users by age and find the total number of users with the role "customer" in each age group.

[{

$match:

{

role:"admin"

}

},

{

$group:{

\_id:"$age",

count:{$sum:1}

}},

{

$project:{

role:"admin",

count:1

}

}

]

28. Group users by role and city, and calculate the total number of users with "DBMS" as a skill in each subgroup.

[{

$match:

{

skills:"DBMS"

}

},

{

$group:{

\_id:{role:"$role",city:"$city"},

count:{$sum:1}

}},

{

$project:{

skills:"DBMS",

count:1

}

}

]  
29. Group users by gender and determine the total number of users with more than three skills in each gender.

30. Group users by city and find the average number of skills for users in each city with salary above 1,000,000.  
31. Group users by age and determine the maximum salary for users with "admin" role in each age group.

32. Group users by role and city, and find the total salary of users with "JavaScript" as a skill in each subgroup.

[{ $match:

{

skills:"JavaScript"

}

},

{

$group:{

\_id:{role:"$role",city:"$city"},

totalsalary:{$sum:"$salary"}

}}]  
33. Group users by gender and determine the average salary for users with the role "customer" in each gender.

[{

$match:

{

role:"customer"

}

},

{

$group:{

\_id:"$gender",

avgsalary:{$avg:"$salary"}

}},

{

$project:{

role:"customer",

avgsalary:1}

}

]  
34. Group users by city and calculate the total number of users with "React" as a skill in each city.

[{

$match:

{

skills:"React"

}

},

{

$group:{

\_id:"$city",

users:{$sum:1}

}}

]  
35. Group users by age and determine the average salary for users with "SQL" as a skill in each age group.

[{

$match:

{

skills:"SQL"

}

},

{

$group:{

\_id:"$age",

averagesalary:{$avg:"$salary"}

}}

]  
36. Group users by role and find the total number of users with "Django" as a skill in each role.

[{

$match:

{

skills:"Django"

}

},

{

$group:{

\_id:"$role",

count:{$sum: 1}

}}

]  
37. Group users by gender and determine the maximum number of skills any user in each gender possesses.

[ {

$group:{

\_id:"$gender",

maxskills:{$max:{$size:"$skills"}}

}}

]  
38. Group users by city and calculate the total number of users with "Node.js" as a skill in each city.

[{

$match:{

skills:"Node.js"

}},

{

$group:{

\_id:"$city",

users:{$sum:1}

}}

]  
39. Group users by age and find the total number of users with "HTML" and "CSS" both as skills in each age group.

[{

$match:{

skills:["HTML","CSS"]

}},

{

$group:{

\_id:"$age",

users:{$sum:1}

}}

]  
40. Group users by role and gender, and determine the average number of skills for each subgroup with salary above 1,000,000.

41. Group users by city and find the average number of users per role in each city.

[

{

$group:{

\_id:{"city":"$city","role":"$role" },

count:{"$sum":1}

}

},

{

$project: {

\_id: "$\_id.city",

avg: {$avg:"$count" }

}

}

]

42. Group users by role and gender, and calculate the total salary for each subgroup.

[{

$group: {

\_id: { "role": "$role", "gender": "$gender" },

totalSalary: { "$sum": "$salary" }

}

}

]

43. Group users by age and determine the total number of users with a salary above 1,000,000.

[

{

$match: {

salary:{"$gt":1000000 }

}

},

{

$group:{

\_id:"$age",

totalUsers:{"$sum":1}

}

}

]

44. Group users by skills and find the average salary for users with each skill.

[{$unwind:"$skills"},

{

$group:{

\_id:"$skills",

avgsalary:{$avg:"$salary"}

}

}

]

45. Group users by role and city, and find the average age for each subgroup.

[

{

$group:{

\_id:{"city":"$city","role":"$role"},

avgage:{$avg:"$age"}

}

}

]

46. Group users by gender and calculate the total number of users with Java as a skill.

[{$unwind:"$skills"},

{

$match:

{skills:"Java"}

},

{

$group:{

\_id:"$gender",

users:{$sum:1}

}

}

]

47. Group users by city and determine the maximum number of skills any user in each city possesses.

[

{$group: {

\_id:"$city",

maxskills:{$max:{$size:"$skills"}}

}

}

]

48. Group users by role and calculate the total number of users for each role with Python as a skill.

[{$unwind:"$skills"},

{

$match:

{skills:"Python"}

},

{

$group:{

\_id:"$role",

users:{$sum:1}

}

}

]

49. Group users by gender and city, and find the average salary for users in each subgroup with more than two skills.

50. Group users by age and determine the average number of skills per user in each age group.

[

{

$group:{

\_id:"$age",

avgnoofskills:{$avg:{$size:"$skills"}}

}}

]

51. Group users by role and find the total salary of users who have "customer" as their role and live in Mumbai.

[{

$match:

{ role:"customer"

,city:"Mumbai"

}

},

{

$group:{

\_id:{role:"$role",city:

"$city"},

salary:{$sum:"$salary"}}

}

]

52. Group users by city and find the maximum salary in each city for users with "admin" role.

[{

$match:

{ role:"admin"

}

},

{

$group:{

\_id:{role:"$role",city:

"$city"},

maxsalary:{$max:"$salary"}}

}

]

53. Group users by gender and calculate the total number of users with salary above 500,000.

54. Group users by skills and determine the average age for users with each skill.

[

{$unwind:"$skills"},

{

$group:{

\_id:"$skills",

avgage:{$avg:"$age"}

}}

]

55. Group users by role and find the average number of skills for users with salary below 700,000.

56. Group users by city and gender, and calculate the average salary for each subgroup with "Python" skill.

[{

$match:

{ skills:"Python"

}

},

{

$group:{

\_id:{gender:"$gender",city:

"$city"},

avgsalary:{$avg:"$salary"}

}}

]

57. Group users by age and determine the total number of users with the role "admin" in each age group.

[{

$match:

{ role:"admin"

}

},

{

$group:{

\_id:{role:"$role",age:

"$age"},

count:{$sum:1}

}}

]

58. Group users by role and city, and find the total number of users with "C++" as a skill in each subgroup.

[{

$match:

{ skills:"C++"

}

},

{

$group:{

\_id:{role:"$role",city:

"$city"},

totalUsers:{$sum:1}

}}

]

59. Group users by gender and calculate the total number of users with salary above the average salary of all users.

[ {

"$match": {

"salary": { "$gt": avgSalary }

}

},{

"$group": {

"\_id": "$gender",

"totalUsers": { "$sum": 1 } }}]

60. Group users by skills and determine the maximum age of users with each skill.

[{$unwind:"$skills"},

{

$group:{

\_id:"$skills",

maxage:{$max:"$age"}

}}

]