1. Group users by age and calculate the average salary for each age group.
ans.

```
db.users.aggregate( [
  {
$group:{
_id: "$age",
averageSal:{$avg:"$salary"}
}
},
  {
    $project: {
     age:"$_id",
      maxSal:1
    }
  }
] )
```

2. Group users by city and find the user with the highest number of skills in each city.
ans.

```
db.users.aggregate( [
{
$group:{
_id: "$city",
maximumSkills:{$max:{"$skills"}}
}
},
  {
    $project: {
      city:"$_id",
      maximumSkills:1
    }
  }
])
```

3. Group users by role and determine the average age for each role.
ans.

```
db.users.aggregate( [
{
$group:{
_id: "$role",
averageAge:{$avg:{"$age"}}
}
```

```
},
  {
    $project: {
      role:"$_id",
      averageAge:1
    }
  }
])
```

4. Group users by skills and calculate the average salary for users with each skill.
ans.

```
db.users.aggregate( [
  {
    $unwind:"$skills"
  },
  {
$group:{
_id: "$skills",
averageSal:{$avg:{"$salary"}}
}
},
  {
    $project: {
     skills:"$_id",
      averageSal:1
    }
  }
])
```

5. Group users by gender and role, and find the total number of users in each subgroup.
ans.

```
db.users.aggregate([
{
$group:{
_id: {gender:"$gender",role:"$role"},
totalUsers:{$sum:1}
}
},
  {
    $project: {
      role:"$_id.role",
      age:"$_id.age",
```

```
      averageSal:1
    }
  }
])
```

6. Group users by city and determine the total number of users in each city.
ans.

```
db.users.aggregate( [
{
$group:{
_id: "$city",
totalUsers:{$sum:1}
}
},
  {
    $project: {
      city:"$_id",
      ToatalUsers:1
    }
  }
])
```

7. Group users by role and calculate the average number of skills per user in each role.
ans.

```
db.users.aggregate( [
{
$project:{
role:1,
NoOfSkills:{$size: "$skills"}
}
},
{
$group:{
_id: "$role",
averageSkills:{$avg:"NoOfSkills"}
}
},
{
$project:{
role: "$_id",
averageSkills:1
}
```

```
}
])
```

8. Group users by gender and find the average salary for each gender.
ans.

```
db.users.aggregate( [
{
$group:{
_id: "$gender",
averageSal:{$avg:{"$salary"}}
}
},
  {
    $project: {
      gender:"$_id",
      averageSal:1
    }
  }
])
```

9. Group users by city and role, and calculate the average age for users in each subgroup.
ans.

```
db.users.aggregate([
{
$group:{
_id: {city:"$city",role:"$role"},
averageAge:{$avg:"$age"}
}
},
  {
    $project: {
      city:"$_id.city",
      role:"$_id.role",
      averageAge:1
    }
  }
])
```

10. Group users by age and find the user with the highest salary in each age group.
ans.

```
db.users.aggregate( [
```

```
{
$group:{
_id: "$age",
maximumSal:{$max:{"$salary"}}
}
},
  {
    $project: {
      age:"$_id",
      maximumSal:1
    }
  }
])
```

11. Group users by role and determine the total number of users in each role.
ans.

```
db.users.aggregate([
$group:{
_id: "$role",
TotalUsers:{$sum:1}
}
},
  {
    $project: {
      role:"$_id",
     TotalUsers:1
    }
  }
])
```

12. Group users by gender and calculate the average number of skills per user in each gender.
ans.

```
db.users.aggregate([
$group:{
_id: "$gender",
averageSkills:{$avg:{"$skills"}},
}
},
  {
    $project: {
      gender:"$_id",
      averageSkills:1
```

```
    }
  }
])
```

13. Group users by city and find the highest and lowest salaries for each city.
ans.

```
db.users.aggregate([
$group:{
_id: "$city",
maximumSal:{$max:{"$salary"}},
minimumSal:{$min:{"$salary"}}
}
},
  {
    $project: {
      city:"$_id",
      maximumSal:1,
      minimumSal:1
    }
  }
])
```

14. Group users by role and age, and calculate the average salary for each subgroup.
ans.

```
db.users.aggregate([
{
$group:{
_id: {role:"$role",age:"$age"},
averageSal:{$avg:"$salary"}
}
},
  {
    $project: {
      role:"$_id.role",
      age:"$_id.age",
      averageSal:1
    }
  }
])
```

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

```
db.users.aggregate( [
  {
    $unwind:"$skills"
  },
  {
$group:{
_id: "$skills",
averageAge:{$avg:{"$age"}}
}
},
  {
    $project: {
     skills:"$_id",
      averageAge:1
    }
  }
]
)
```