DAY 1

```
create database db;
use db;

-- 1
create table department(
dep_id int,
dep_name varchar(20),
dep_loc varchar(15),
primary key (dep_id)
);

create table employees(
emp_id int,
emp_name varchar(15),
job_name varchar(10),
manager_id int,
hire_date date,
salary decimal(10,2),
commission decimal(7,2),
dep_id int,
primary key (emp_id),
foreign key(dep_id) references department(dep_id)
on delete cascade
on update cascade
);

insert into department values
(1,'HR','Bangalore'),
(2,'Software','Mysore'),
(3,'Accounts','Hassan');

insert into employees values
(101, 'John Smith', 'Manager', NULL, '1991-02-22', 60000.00, NULL, 1),
(102, 'Jane Doe', 'Analyst', 101, '1991-02-22', 45000.00, 500.00, 1),
(103, 'Michael Johnson', 'Manager', NULL, '1991-02-22', 65000.00, NULL, 2),
(104, 'Emily Brown', 'Clerk', 103, '1991-02-22', 55000.00, 700.00, 2),
(105, 'David Lee', 'Clerk', 103, '2021-07-15', 70000.00, 1000.00, 3);

-- 2
select salary from employees;
```

Result Grid | Filter F

salary
60000.00
45000.00
65000.00
55000.00
70000.00

```
-- 3
select emp_name
from employees
where hire_date = '1991-02-22';
```

Result Grid | Filter Rows:

emp_name
John Smith
Jane Doe
Michael Johnson
Emily Brown

```
-- 4
select avg(salary)
from employees
where job_name='Analyst';
```

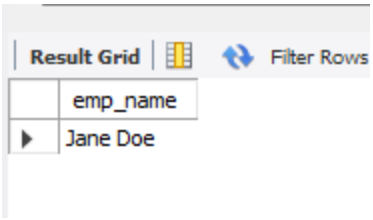
avg(salary)
45000.000000

```
-- 5
select emp_name
from employees
where job_name='Clerk' or job_name='Manager';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

emp_name
John Smith
Michael Johnson
Emily Brown
David Lee

```
-- 6
select emp_name
from employees
where salary between 24000 and 50000;
```



26/03/2024 **DAY 2**

I. Perform the following DB operations using MongoDB.

1. Create a database “Student” with the following attributes Rollno, Age, ContactNo, Email-Id.

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> use myDB;
already on db myDB
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.createCollection("Student");
{ ok: 1 }
```

2. Insert appropriate values

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.insert({rollno:11,age:20,phno:9887645328,email:"abc@gmail.com",name:"ABC"})
DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMany, or bulkWrite.
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("66028ff3b495b09bc2b90d34") }
}
```

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.insert({rollno:10,age:21,phno:9848574328,email:"efg@gmail.com",name:"EFG"})
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6602902db495b09bc2b90d35") }
}
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.insert({rollno:12,age:21,phno:8748574328,email:"hij@gmail.com",name:"HIJ"})
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6602908db495b09bc2b90d36") }
}
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.insert({rollno:13,age:23,phno:7748574328,email:"riya@gmail.com",name:"Riya"})
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("660243e7b495b09bc2b90d37") }
}
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.insert({rollno:14,age:22,phno:7493574395,email:"pihu@gmail.com",name:"Pihu"})
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6602440eb495b09bc2b90d38") }
}
```

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.find();
[
  {
    _id: ObjectId("66028ff3b495b09bc2b90d34"),
    rollno: 11,
    age: 20,
    phno: 9887645328,
    email: 'abc@gmail.com',
    name: 'ABC'
  },
  {
    _id: ObjectId("6602902db495b09bc2b90d35"),
    rollno: 10,
    age: 21,
    phno: 9848574328,
    email: 'efg@gmail.com',
    name: 'EFG'
  },
  {
    _id: ObjectId("6602908db495b09bc2b90d36"),
    rollno: 12,
    age: 21,
    phno: 8748574328,
    email: 'hij@gmail.com',
    name: 'HIJ'
  },
  {
    _id: ObjectId("660243e7b495b09bc2b90d37"),
    rollno: 13,
    age: 23,
    phno: 7748574328,
    email: 'riya@gmail.com',
    name: 'Riya'
  },
  {
    _id: ObjectId("6602440eb495b09bc2b90d38"),
    rollno: 14,
    age: 22,
    phno: 7493574395,
    email: 'pihu@gmail.com',
    name: 'Pihu'
  }
]
```

3. Write query to update Email-Id of a student with rollno 10.

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.update({rollno:10},{$set:{email:"efg1@gmail.com"}});
DeprecationWarning: Collection.update() is deprecated. Use updateOne, updateMany, or bulkWrite.
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.find();
[
  {
    _id: ObjectId("66028ff3b495b09bc2b90d34"),
    rollno: 11,
    age: 20,
    phno: 9887645328,
    email: 'abc@gmail.com',
    name: 'ABC'
  },
  {
    _id: ObjectId("6602902db495b09bc2b90d35"),
    rollno: 10,
    age: 21,
    phno: 9848574328,
    email: 'efg1@gmail.com',
    name: 'EFG'
  },
  {
    _id: ObjectId("6602908db495b09bc2b90d36"),
    rollno: 12,
    age: 21,
    phno: 8748574328,
    email: 'hij@gmail.com',
    name: 'HIJ'
  },
  {
    _id: ObjectId("660243e7b495b09bc2b90d37"),
    rollno: 13,
    age: 23,
    phno: 7748574328,
    email: 'riya@gmail.com',
    name: 'Riya'
  },
  {
    _id: ObjectId("6602440eb495b09bc2b90d38"),
    rollno: 14,
    age: 22,
    phno: 7493574395,
    email: 'pihu@gmail.com',
    name: 'Pihu'
  }
]
```

4. . Replace the student name from “ABC” to “FEM” of rollno 11

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.update({rollno:11},{$set:{name:"FEM"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.find();
[
  {
    _id: ObjectId("66028ff3b495b09bc2b90d34"),
    rollno: 11,
    age: 20,
    phno: 9887645328,
    email: 'abc@gmail.com',
    name: 'FEM'
  },
  {
    _id: ObjectId("6602902db495b09bc2b90d35"),
    rollno: 10,
    age: 21,
    phno: 9848574328,
    email: 'efg1@gmail.com',
    name: 'EFG'
  },
  {
    _id: ObjectId("6602908db495b09bc2b90d36"),
    rollno: 12,
    age: 21,
    phno: 8748574328,
    email: 'hij@gmail.com',
    name: 'HIJ'
  },
  {
    _id: ObjectId("660243e7b495b09bc2b90d37"),
    rollno: 13,
    age: 23,
    phno: 7748574328,
    email: 'riya@gmail.com',
    name: 'Riya'
  },
  {
    _id: ObjectId("6602440eb495b09bc2b90d38"),
    rollno: 14,
    age: 22,
    phno: 7493574395,
    email: 'pihu@gmail.com',
    name: 'Pihu'
  }
]
```

II. Perform the following DB operations using MongoDB.

1. Create a collection by name Customers with the following attributes.
Cust_id, Acc_Bal, Acc_Type

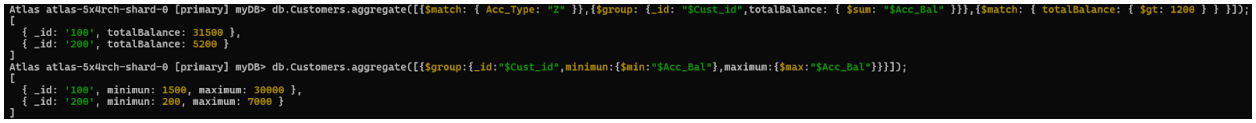
```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.createCollection("Customers");
{ ok: 1 }
```

2. Insert at least 5 values into the table

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Customers.insert({Cust_id:"100",Acc_Bal:1500,Acc_Type:"Z"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("66024cefb495b09bc2b90d40") }
}
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Customers.insert({Cust_id:"200",Acc_Bal:7000,Acc_Type:"A"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("66024cfeb495b09bc2b90d41") }
}
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Customers.insert({Cust_id:"200",Acc_Bal:200,Acc_Type:"Z"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("66024d0bb495b09bc2b90d42") }
}
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Customers.insert({Cust_id:"100",Acc_Bal:30000,Acc_Type:"Z"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("66024d29b495b09bc2b90d43") }
}
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Customers.insert({Cust_id:"200",Acc_Bal:5000,Acc_Type:"Z"});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("66024d3eb495b09bc2b90d44") }
}
```

3. Write a query to display those records whose total account balance is greater than 1200 of account type ‘Z’ for each customer_id.

4. Determine Minimum and Maximum account balance for each customer_i



DAY 3

I. Perform the following DB operations using MongoDB.

5. Display Student Name and grade(Add if grade is not present)where the _id column is 1.

db.Student.updateMany({},{ $set: { "grade": "A" } })

db.Student.find({"rollno":11},{"name":1,"grade":1});

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.updateMany({},{ $set: { "grade": "A" } })
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 5,
  modifiedCount: 5,
  upsertedCount: 0
}
```

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.find();
[
  {
    _id: ObjectId("66028ff3b495b09bc2b90d34"),
    rollno: 11,
    age: 20,
    phno: 9887645328,
    email: 'abc@gmail.com',
    name: 'FEM',
    grade: 'A'
  },
  {
    _id: ObjectId("6602902db495b09bc2b90d35"),
    rollno: 10,
    age: 21,
    phno: 9848574328,
    email: 'efg1@gmail.com',
    name: 'EFG',
    grade: 'A'
  },
  {
    _id: ObjectId("6602908db495b09bc2b90d36"),
    rollno: 12,
    age: 21,
    phno: 8748574328,
    email: 'hij@gmail.com',
    name: 'HIJ',
    grade: 'A'
  },
  {
    _id: ObjectId("660243e7b495b09bc2b90d37"),
    rollno: 13,
    age: 23,
    phno: 7748574328,
    email: 'riya@gmail.com',
    name: 'Riya',
    grade: 'A'
  },
  {
    _id: ObjectId("6602440eb495b09bc2b90d38"),
    rollno: 14,
    age: 22,
    phno: 7493574395,
    email: 'pihu@gmail.com',
    name: 'Pihu',
    grade: 'A'
  }
]
```

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.find({"rollno":11},{"name":1,"grade":1});
[
  {
    _id: ObjectId("66028ff3b495b09bc2b90d34"),
    name: 'FEM',
    grade: 'A'
  }
]
```

6. Update to add hobbies

```
db.Student.update({"rollno":10},{$set:{"hobbies":"chess"}});
db.Student.update({"rollno":11},{$set:{"hobbies":"skating"}});
db.Student.update({"rollno":12},{$set:{"hobbies":"singing"}});
db.Student.update({"rollno":13},{$set:{"hobbies":"cricket"}});
db.Student.update({"rollno":14},{$set:{"hobbies":"painting"}});
```

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.update({"rollno":10},{$set:{"hobbies":"chess"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.update({"rollno":11},{$set:{"hobbies":"skating"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.update({"rollno":12},{$set:{"hobbies":"singing"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.update({"rollno":13},{$set:{"hobbies":"cricket"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.update({"rollno":14},{$set:{"hobbies":"painting"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

7. Find documents where hobbies is set neither to Chess nor to Skating

```
db.Student.find({"hobbies": {$nin: ["chess", "skating"] }})
```

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.find({"hobbies": {$nin: ["chess", "skating"] }})
[
  {
    _id: ObjectId("6602908db495b09bc2b90d36"),
    rollno: 12,
    age: 21,
    phno: 8748574328,
    email: 'hij@gmail.com',
    name: 'HIJ',
    grade: 'A',
    hobbies: 'singing'
  },
  {
    _id: ObjectId("660243e7b495b09bc2b90d37"),
    rollno: 13,
    age: 23,
    phno: 7748574328,
    email: 'riya@gmail.com',
    name: 'Riya',
    grade: 'A',
    hobbies: 'cricket'
  },
  {
    _id: ObjectId("6602440eb495b09bc2b90d38"),
    rollno: 14,
    age: 22,
    phno: 7493574395,
    email: 'pihu@gmail.com',
    name: 'Pihu',
    grade: 'A',
    hobbies: 'painting'
  }
]
```

8.Find documents whose name begins with A

db.Student.find({"name": /^A/})

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.insert({rollno:15,age:23,phno:7744567428,email:"aero@gmail.com",name:"Aero"}
DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMany, or bulkWrite.
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("660bd039aad18859618dfa6d") }
}
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Student.find({"name":/^A/});
[
  {
    _id: ObjectId("660bd039aad18859618dfa6d"),
    rollno: 15,
    age: 23,
    phno: 7744567428,
    email: 'aero@gmail.com',
    name: 'Aero'
  }
]
```

II. Perform the following DB operations using MongoDB.

5. Sort the documents based on Customer ID in ascending order and Account Balance in descending order

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Customers.find().sort({"Cus_id":1,"Acc_Bal":-1}).pretty();
[
  {
    _id: ObjectId("66024d29b495b09bc2b90d43"),
    Cust_id: '100',
    Acc_Bal: 30000,
    Acc_Type: 'Z'
  },
  {
    _id: ObjectId("66024cfeb495b09bc2b90d41"),
    Cust_id: '200',
    Acc_Bal: 7000,
    Acc_Type: 'A'
  },
  {
    _id: ObjectId("66024d3eb495b09bc2b90d44"),
    Cust_id: '200',
    Acc_Bal: 5000,
    Acc_Type: 'Z'
  },
  {
    _id: ObjectId("66024cefb495b09bc2b90d40"),
    Cust_id: '100',
    Acc_Bal: 1500,
    Acc_Type: 'Z'
  },
  {
    _id: ObjectId("66024d0bb495b09bc2b90d42"),
    Cust_id: '200',
    Acc_Bal: 200,
    Acc_Type: 'Z'
  }
]
```

6. Display only 2 nd and 3 rd records from the collection

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.Customers.find().pretty().skip(1).limit(2);
[
  {
    _id: ObjectId("66024cfeb495b09bc2b90d41"),
    Cust_id: '200',
    Acc_Bal: 7000,
    Acc_Type: 'A'
  },
  {
    _id: ObjectId("66024d0bb495b09bc2b90d42"),
    Cust_id: '200',
    Acc_Bal: 200,
    Acc_Type: 'Z'
  }
]
```

III. Perform the following DB operations using MongoDB

Create a collection by the name blogPosts and it has 3 fields id, title and comments.
In the collection the comments field is an array which consists of user details. Each collection consists of two user details inside the comments array- user name and text
Demonstrate the following

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.blogPost.insertMany([{id:1, title:"Blog 1", comments:["Nice blog","Worst blog
ever","Very concise blog"]},{id:2, title:"Blog 2", comments:["Mediocre","Very detailed information","Informative and enterta
ining"]},{id:3, title:"Blog 3", comments:["Awesome","Detailed information","Informative but boring"]},{id:4, title:"Blog 4",
 comments:["Sikkkeeeee","useless","lolll, lmao"]},{id:5, title:"Blog 5", comments:["very nice","Very cool information","Info
rmative and entertaining"]}]);
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("660bd77baad18859618dfa6e"),
    '1': ObjectId("660bd77baad18859618dfa6f"),
    '2': ObjectId("660bd77baad18859618dfa70"),
    '3': ObjectId("660bd77baad18859618dfa71"),
    '4': ObjectId("660bd77baad18859618dfa72")
  }
}
```

1. Adding an element into araay

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.blogPost.update({id:1},{$push:{comments:"User 1"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-5x4rch-shard-0 [primary] myDB>  db.blogPost.update({id:2},{$push:{comments:"User 2"}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.blogPost.update({id:3},{$push:{comments:"User 3"}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-5x4rch-shard-0 [primary] myDB>     db.blogPost.update({id:4},{$push:{comments:"User 4"}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Atlas atlas-5x4rch-shard-0 [primary] myDB>   db.blogPost.update({id:5},{$push:{comments:"User 5"}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

2. Display second element

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.blogPost.aggregate([{$project:{secondElement:{$arrayElemAt:["$comments",1]}}}]
)
[
  {
    _id: ObjectId("660bd77baad18859618dfa6e"),
    secondElement: 'Worst blog ever'
  },
  {
    _id: ObjectId("660bd77baad18859618dfa6f"),
    secondElement: 'Very detailed information'
  },
  {
    _id: ObjectId("660bd77baad18859618dfa70"),
    secondElement: 'Detailed information'
  },
  {
    _id: ObjectId("660bd77baad18859618dfa71"),
    secondElement: 'useless'
  },
  {
    _id: ObjectId("660bd77baad18859618dfa72"),
    secondElement: 'Very cool information'
  }
]
```

3. Display size of the array

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.blogPost.aggregate([{$project:{arraySize:{$size:"$comments"}}}])
[
  { _id: ObjectId("660bd77baad18859618dfa6e"), arraySize: 4 },
  { _id: ObjectId("660bd77baad18859618dfa6f"), arraySize: 4 },
  { _id: ObjectId("660bd77baad18859618dfa70"), arraySize: 4 },
  { _id: ObjectId("660bd77baad18859618dfa71"), arraySize: 4 },
  { _id: ObjectId("660bd77baad18859618dfa72"), arraySize: 4 }
]
```

4. Display first two elements of the array

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.blogPost.aggregate([{$project:{secondElement:{$arrayElemAt:["$comments",1]}}}]
)
[
  {
    _id: ObjectId("660bd77baad18859618dfa6e"),
    secondElement: 'Worst blog ever'
  },
  {
    _id: ObjectId("660bd77baad18859618dfa6f"),
    secondElement: 'Very detailed information'
  },
  {
    _id: ObjectId("660bd77baad18859618dfa70"),
    secondElement: 'Detailed information'
  },
  {
    _id: ObjectId("660bd77baad18859618dfa71"),
    secondElement: 'useless'
  },
  {
    _id: ObjectId("660bd77baad18859618dfa72"),
    secondElement: 'Very cool information'
  }
]
```

5. Update the document with id 4 and replace the element present in 1st index position of the array with another array

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.blogPost.update({id:4},{$set:{"comments.1":["hello","nice blog"]}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

```
Atlas atlas-5x4rch-shard-0 [primary] myDB> db.blogPost.find();
[
  {
    _id: ObjectId("660bd77baad18859618dfa6e"),
    id: 1,
    title: 'Blog 1',
    comments: [ 'Nice blog', 'Worst blog ever', 'Very concise blog', 'User 1' ]
  },
  {
    _id: ObjectId("660bd77baad18859618dfa6f"),
    id: 2,
    title: 'Blog 2',
    comments: [
      'Mediocre',
      'Very detailed information',
      'Informative and entertaining',
      'User 2'
    ]
  },
  {
    _id: ObjectId("660bd77baad18859618dfa70"),
    id: 3,
    title: 'Blog 3',
    comments: [
      'Awesome',
      'Detailed information',
      'Informative but boring',
      'User 3'
    ]
  },
  {
    _id: ObjectId("660bd77baad18859618dfa71"),
    id: 4,
    title: 'Blog 4',
    comments: [ 'Sikkkeeeee', [ 'hello', 'nice blog' ], 'lolll, lmao', 'User 4' ]
  },
  {
    _id: ObjectId("660bd77baad18859618dfa72"),
    id: 5,
    title: 'Blog 5',
    comments: [
      'very nice',
      'Very cool information',
      'Informative and entertaining',
      'User 5'
    ]
  }
]
```