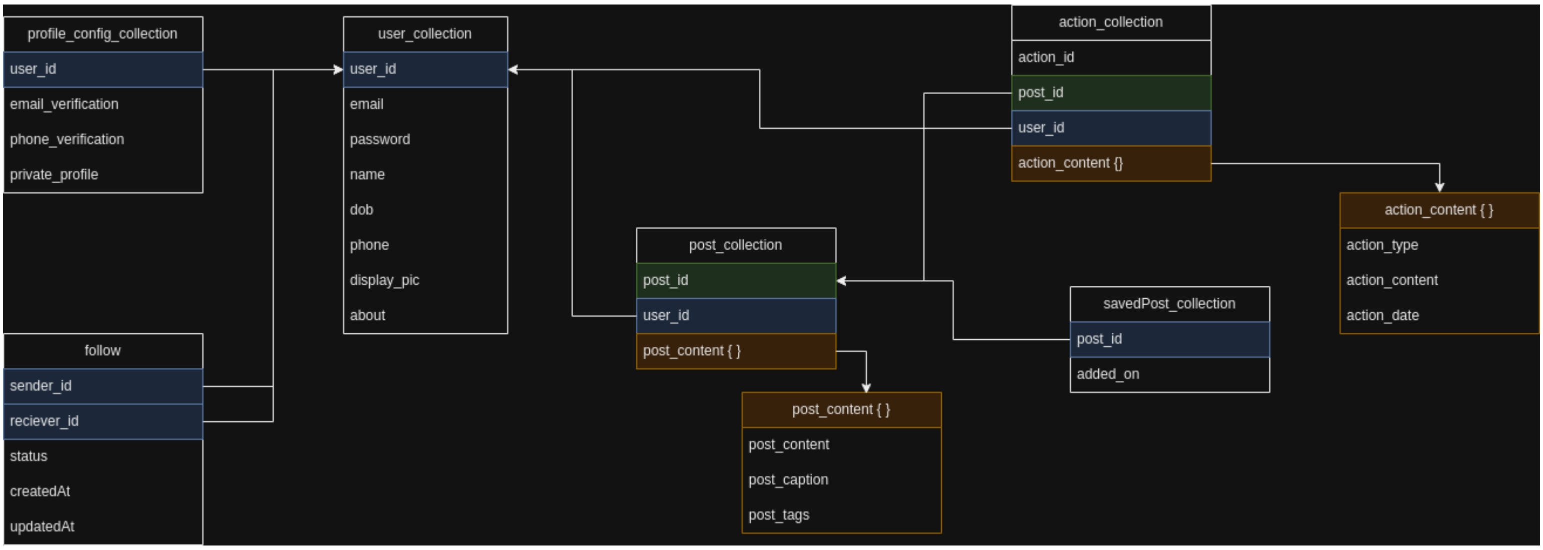
MongoDB Assigment

Schema for instagram :



profile_config_collection
- user_id
- email_verification
- phone_verification
- private_profile

user_collection
- user_id
- email
- password
- name
- dob
- phone
- display_pic
- about

follow
- sender_id
- reciever_id
- status
- createdAt
- updatedAt

post_collection
- post_id
- user_id
- post_content { }

post_content { }
- post_content
- post_caption
- post_tags

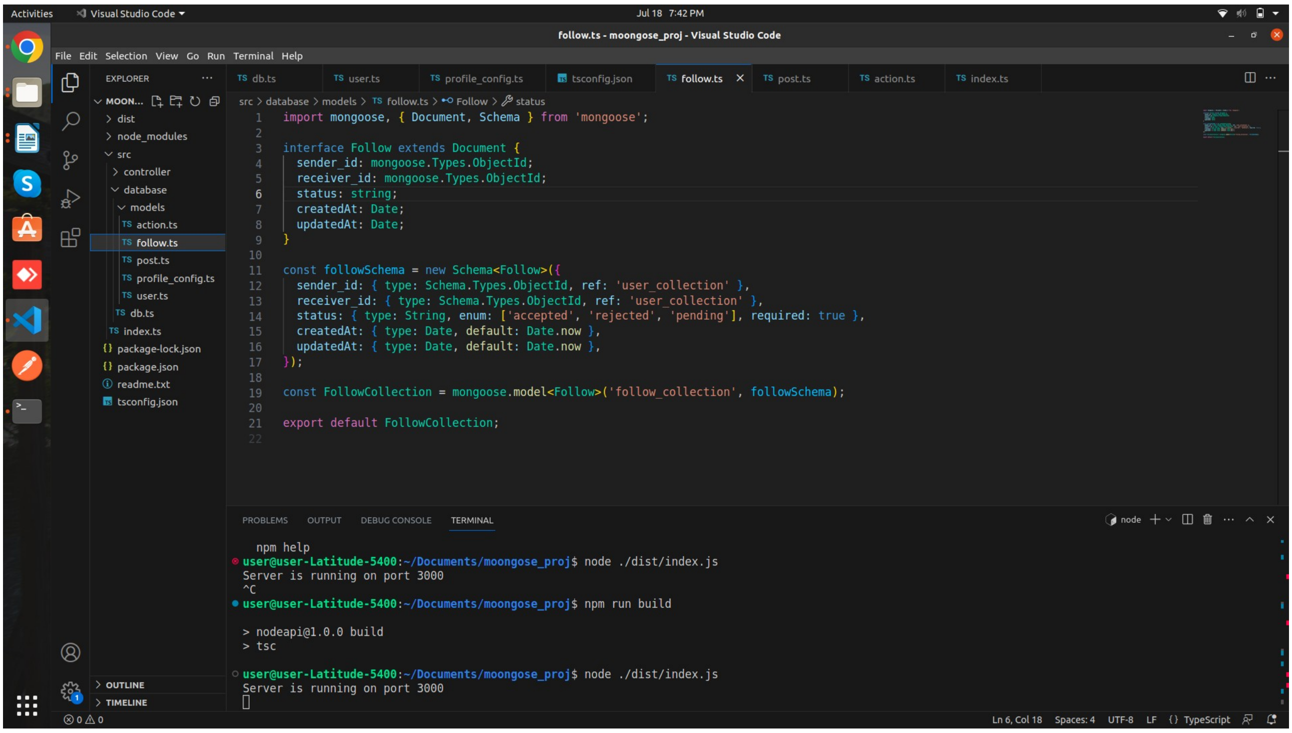
action_collection
- action_id
- post_id
- user_id
- action_content {}

action_content { }
- action_type
- action_content
- action_date

savedPost_collection
- post_id
- added_on

Models in moongose:

follow_model

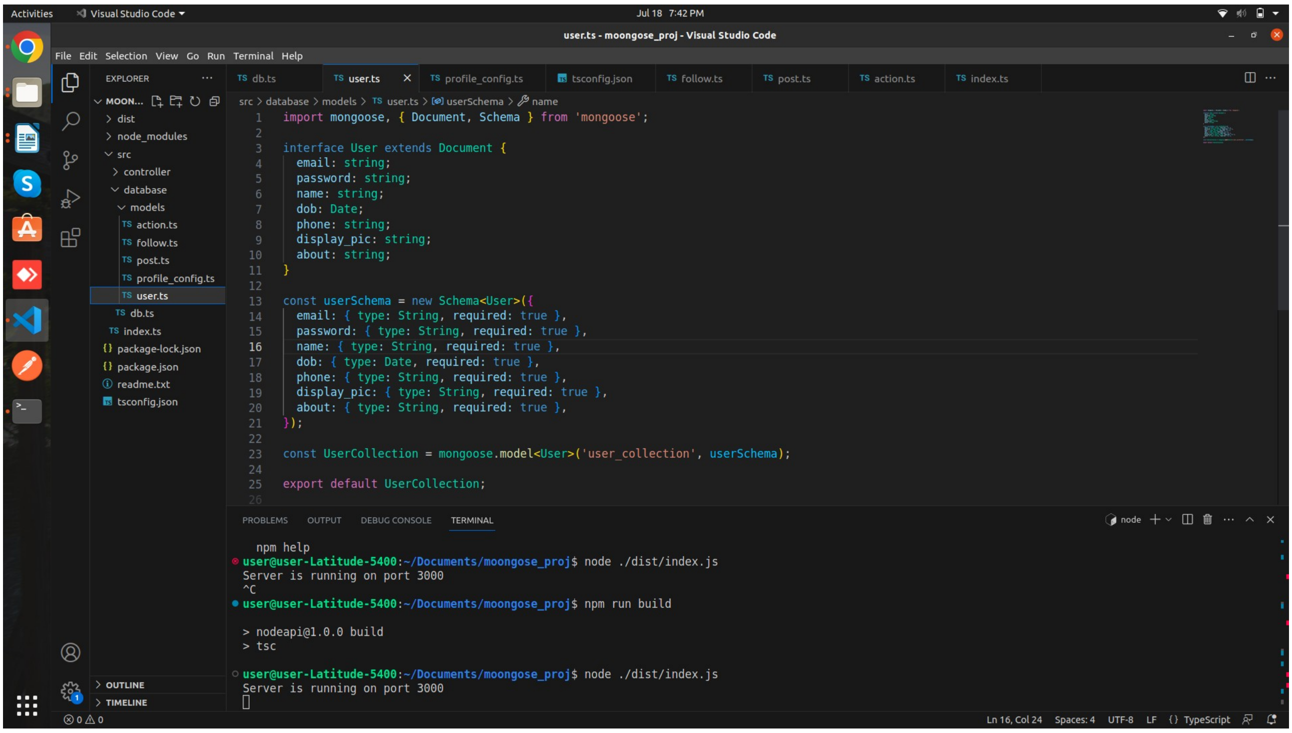


follow.ts - moongose_proj - Visual Studio Code
db.ts
user.ts
profile_config.ts
tsconfig.json
follow.ts
post.ts
action.ts
index.ts
src > database > models > follow.ts > Follow > status

```typescript
import mongoose, { Document, Schema } from 'mongoose';

interface Follow extends Document {
  sender_id: mongoose.Types.ObjectId;
  receiver_id: mongoose.Types.ObjectId;
  status: string;
  createdAt: Date;
  updatedAt: Date;
}

const followSchema = new Schema<Follow>({
  sender_id: { type: Schema.Types.ObjectId, ref: 'user_collection' },
  receiver_id: { type: Schema.Types.ObjectId, ref: 'user_collection' },
  status: { type: String, enum: ['accepted', 'rejected', 'pending'], required: true },
  createdAt: { type: Date, default: Date.now },
  updatedAt: { type: Date, default: Date.now },
});

const FollowCollection = mongoose.model<Follow>('follow_collection', followSchema);

export default FollowCollection;
```

```
  npm help
user@user-Latitude-5400:~/Documents/moongose_proj$ node ./dist/index.js
Server is running on port 3000
^C
user@user-Latitude-5400:~/Documents/moongose_proj$ npm run build

> nodeapi@1.0.0 build
> tsc

user@user-Latitude-5400:~/Documents/moongose_proj$ node ./dist/index.js
Server is running on port 3000
```

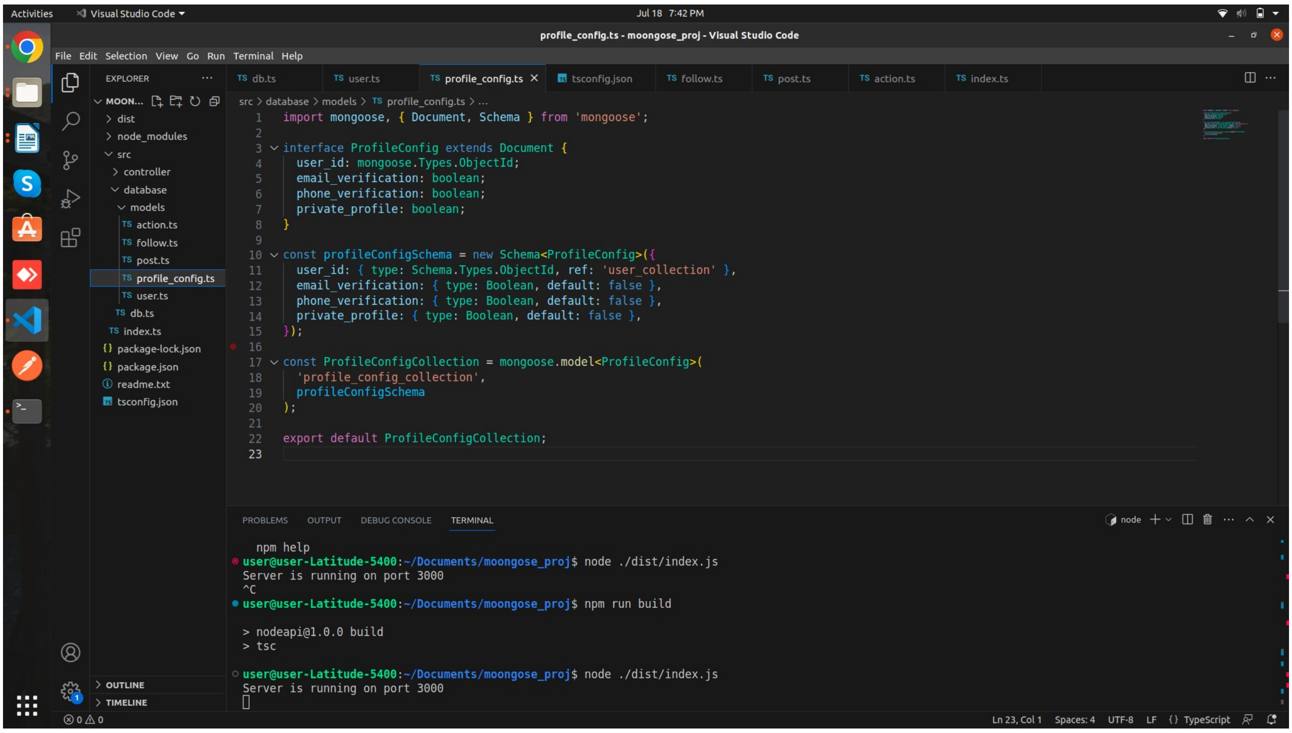
user_model



```typescript
import mongoose, { Document, Schema } from 'mongoose';

interface User extends Document {
  email: string;
  password: string;
  name: string;
  dob: Date;
  phone: string;
  display_pic: string;
  about: string;
}

const userSchema = new Schema<User>({
  email: { type: String, required: true },
  password: { type: String, required: true },
  name: { type: String, required: true },
  dob: { type: Date, required: true },
  phone: { type: String, required: true },
  display_pic: { type: String, required: true },
  about: { type: String, required: true },
});

const UserCollection = mongoose.model<User>('user_collection', userSchema);

export default UserCollection;
```

```
  npm help
user@user-Latitude-5400:~/Documents/moongose_proj$ node ./dist/index.js
Server is running on port 3000
^C
user@user-Latitude-5400:~/Documents/moongose_proj$ npm run build

> nodeapi@1.0.0 build
> tsc

user@user-Latitude-5400:~/Documents/moongose_proj$ node ./dist/index.js
Server is running on port 3000
```

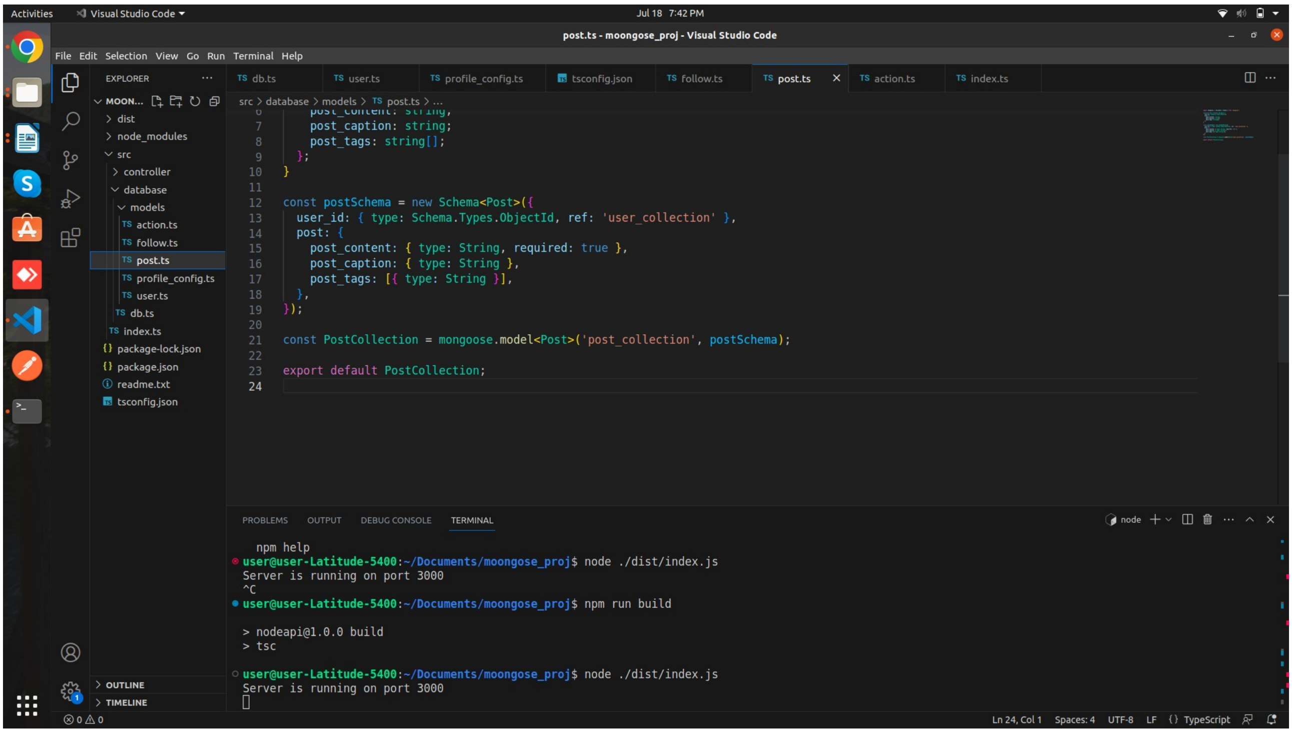
profile_config_model



```typescript
import mongoose, { Document, Schema } from 'mongoose';

interface ProfileConfig extends Document {
  user_id: mongoose.Types.ObjectId;
  email_verification: boolean;
  phone_verification: boolean;
  private_profile: boolean;
}

const profileConfigSchema = new Schema<ProfileConfig>({
  user_id: { type: Schema.Types.ObjectId, ref: 'user_collection' },
  email_verification: { type: Boolean, default: false },
  phone_verification: { type: Boolean, default: false },
  private_profile: { type: Boolean, default: false },
});

const ProfileConfigCollection = mongoose.model<ProfileConfig>(
  'profile_config_collection',
  profileConfigSchema
);

export default ProfileConfigCollection;

```

```
  npm help
user@user-Latitude-5400:~/Documents/moongose_proj$ node ./dist/index.js
Server is running on port 3000
^C
user@user-Latitude-5400:~/Documents/moongose_proj$ npm run build

> nodeapi@1.0.0 build
> tsc

user@user-Latitude-5400:~/Documents/moongose_proj$ node ./dist/index.js
Server is running on port 3000
```

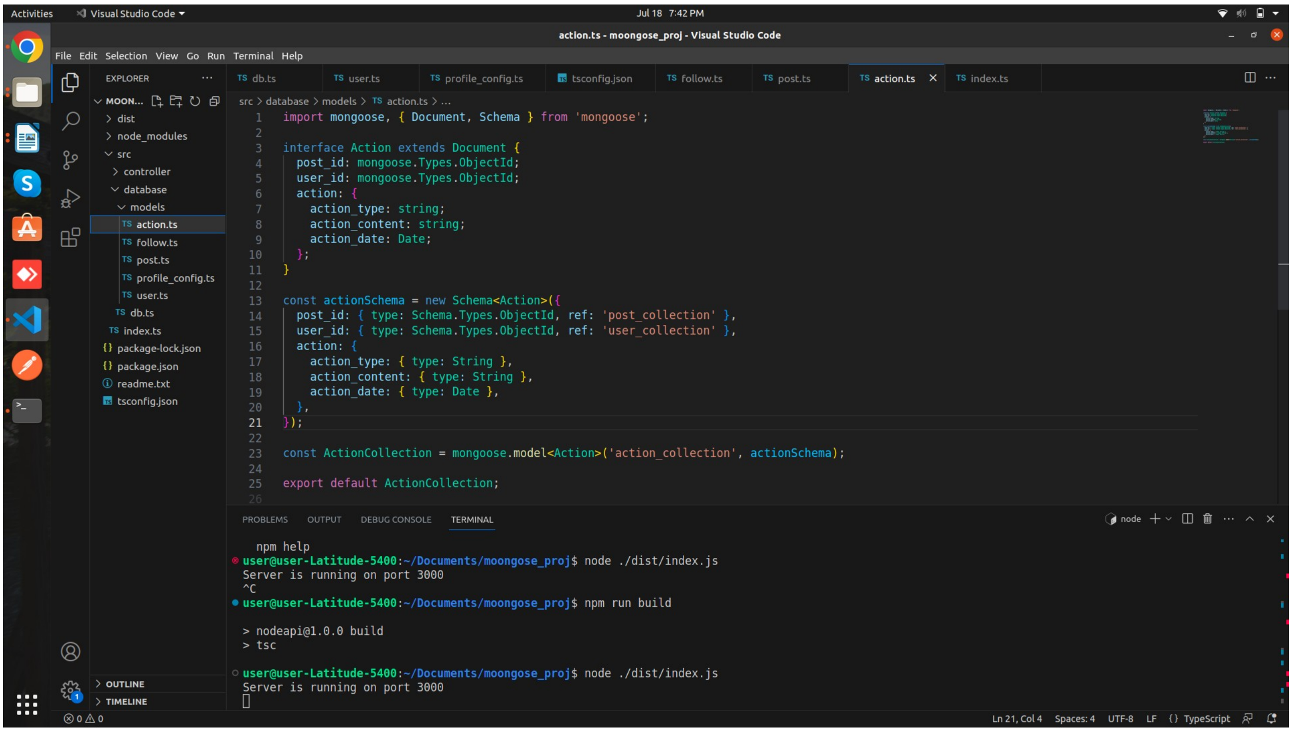
post_model



```
    post_content: string;
    post_caption: string;
    post_tags: string[];
  };
}

const postSchema = new Schema<Post>({
  user_id: { type: Schema.Types.ObjectId, ref: 'user_collection' },
  post: {
    post_content: { type: String, required: true },
    post_caption: { type: String },
    post_tags: [{ type: String }],
  },
});

const PostCollection = mongoose.model<Post>('post_collection', postSchema);

export default PostCollection;
```

```
  npm help
user@user-Latitude-5400:~/Documents/moongose_proj$ node ./dist/index.js
Server is running on port 3000
^C
user@user-Latitude-5400:~/Documents/moongose_proj$ npm run build

> nodeapi@1.0.0 build
> tsc

user@user-Latitude-5400:~/Documents/moongose_proj$ node ./dist/index.js
Server is running on port 3000
```

action_model

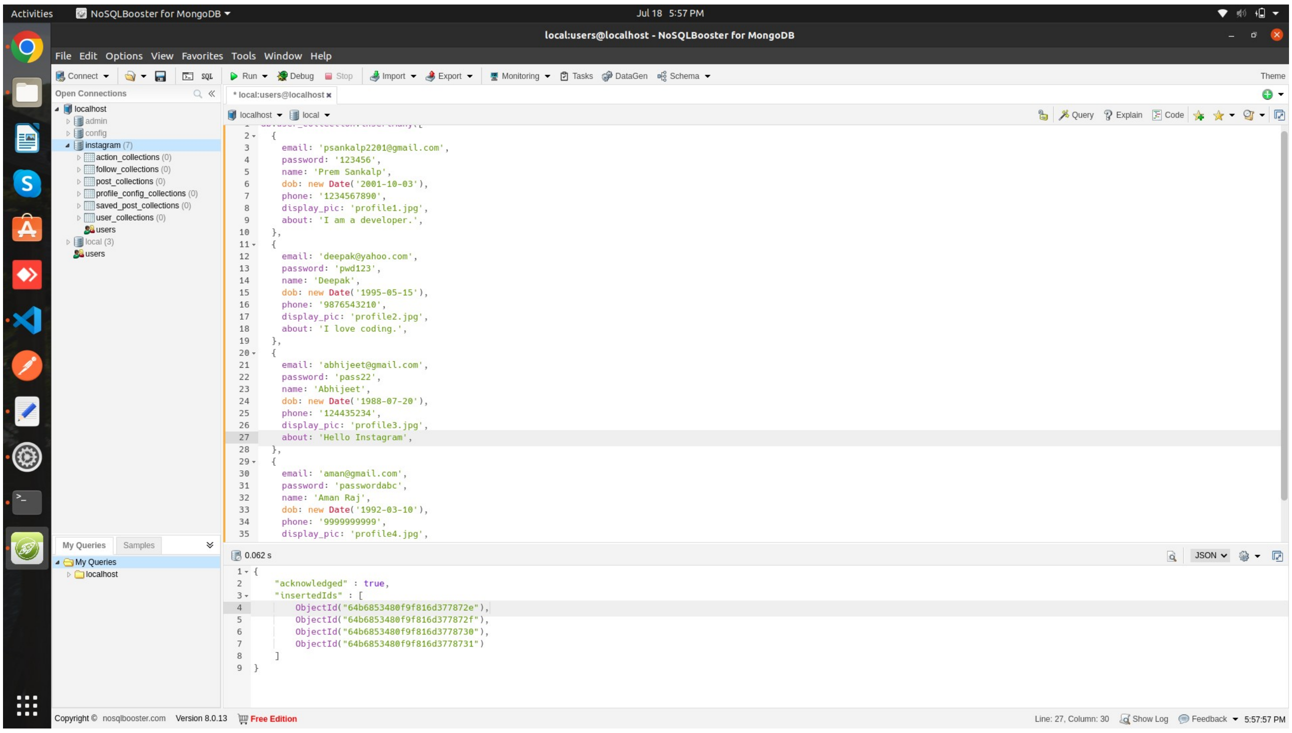


```
import mongoose, { Document, Schema } from 'mongoose';

interface Action extends Document {
  post_id: mongoose.Types.ObjectId;
  user_id: mongoose.Types.ObjectId;
  action: {
    action_type: string;
    action_content: string;
    action_date: Date;
  };
}

const actionSchema = new Schema<Action>({
  post_id: { type: Schema.Types.ObjectId, ref: 'post_collection' },
  user_id: { type: Schema.Types.ObjectId, ref: 'user_collection' },
  action: {
    action_type: { type: String },
    action_content: { type: String },
    action_date: { type: Date },
  },
});

const ActionCollection = mongoose.model<Action>('action_collection', actionSchema);

export default ActionCollection;
```

```
  npm help
user@user-Latitude-5400:~/Documents/moongose_proj$ node ./dist/index.js
Server is running on port 3000
^C
user@user-Latitude-5400:~/Documents/moongose_proj$ npm run build

> nodeapi@1.0.0 build
> tsc

user@user-Latitude-5400:~/Documents/moongose_proj$ node ./dist/index.js
Server is running on port 3000
```

Queries in nosqlbooster

Inserton on user

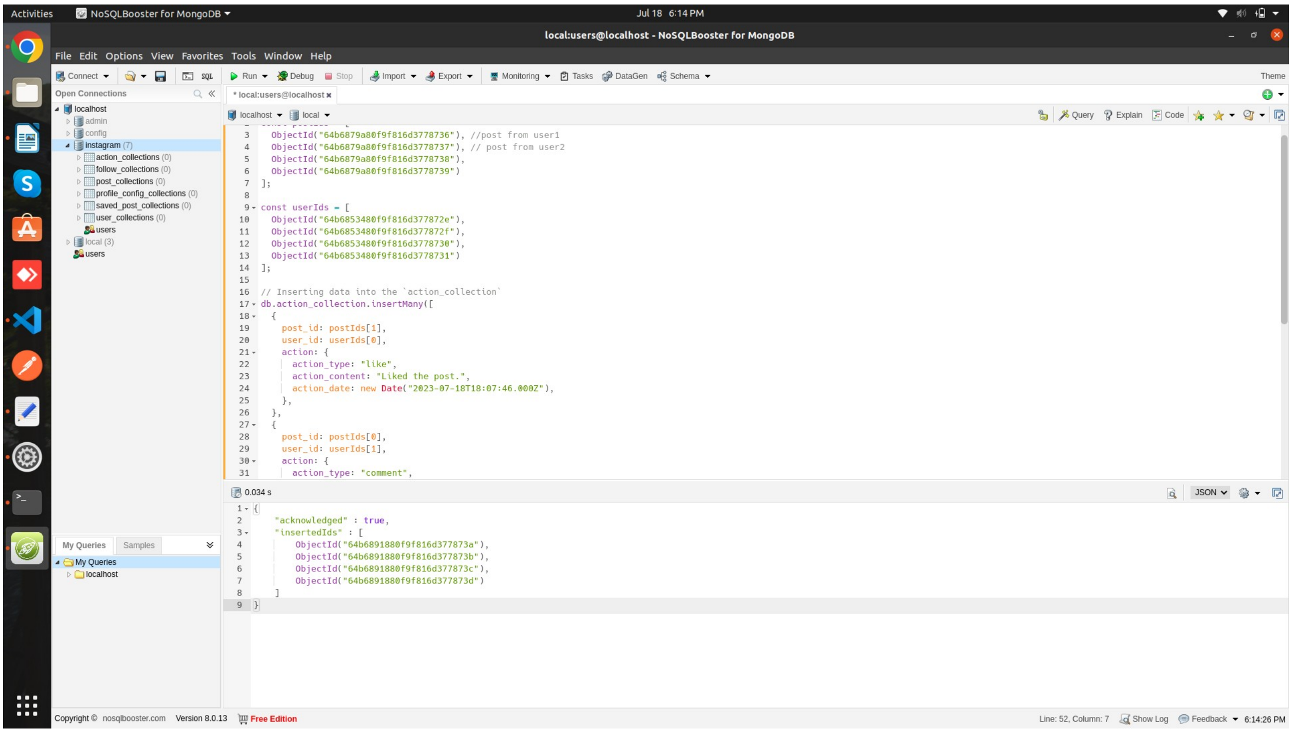


```
db.user_collection.insertMany([
  {
    email: 'psankalp2201@gmail.com',
    password: '123456',
    name: 'Prem Sankalp',
    dob: new Date('2001-10-03'),
    phone: '1234567890',
    display_pic: 'profile1.jpg',
    about: 'I am a developer.',
  },
  {
    email: 'deepak@yahoo.com',
    password: 'pwd123',
    name: 'Deepak',
    dob: new Date('1995-05-15'),
    phone: '9876543210',
    display_pic: 'profile2.jpg',
    about: 'I love coding.',
  },
  {
    email: 'abhijeet@gmail.com',
    password: 'pass22',
    name: 'Abhijeet',
    dob: new Date('1988-07-20'),
    phone: '124435234',
    display_pic: 'profile3.jpg',
    about: 'Hello Instagram',
  },
  {
    email: 'aman@gmail.com',
    password: 'passwordabc',
    name: 'Aman Raj',
    dob: new Date('1992-03-10'),
    phone: '9999999999',
    display_pic: 'profile4.jpg',
```

0.062 s

```
{
    "acknowledged" : true,
    "insertedIds" : [
        ObjectId("64b6853480f9f816d377872e"),
        ObjectId("64b6853480f9f816d377872f"),
        ObjectId("64b6853480f9f816d3778730"),
        ObjectId("64b6853480f9f816d3778731")
    ]
}
```

insertion on post_action

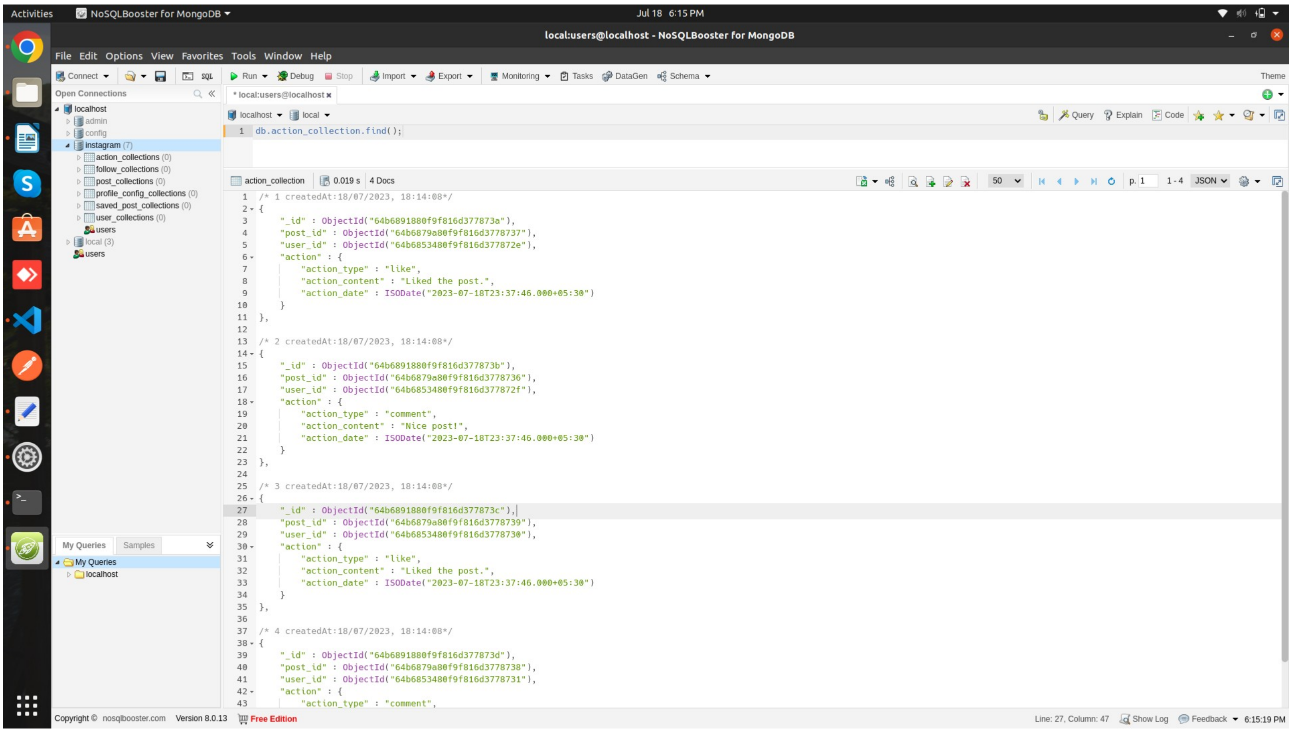


```
const postIds = [
  ObjectId("64b6879a80f9f816d3778736"), //post from user1
  ObjectId("64b6879a80f9f816d3778737"), // post from user2
  ObjectId("64b6879a80f9f816d3778738"),
  ObjectId("64b6879a80f9f816d3778739")
];

const userIds = [
  ObjectId("64b6853480f9f816d377872e"),
  ObjectId("64b6853480f9f816d377872f"),
  ObjectId("64b6853480f9f816d3778730"),
  ObjectId("64b6853480f9f816d3778731")
];

// Inserting data into the `action_collection`
db.action_collection.insertMany([
  {
    post_id: postIds[1],
    user_id: userIds[0],
    action: {
      action_type: "like",
      action_content: "Liked the post.",
      action_date: new Date("2023-07-18T18:07:46.000Z"),
    },
  },
  {
    post_id: postIds[0],
    user_id: userIds[1],
    action: {
      action_type: "comment",
```

0.034 s

```
{
    "acknowledged" : true,
    "insertedIds" : [
        ObjectId("64b6891880f9f816d377873a"),
        ObjectId("64b6891880f9f816d377873b"),
        ObjectId("64b6891880f9f816d377873c"),
        ObjectId("64b6891880f9f816d377873d")
    ]
}
```

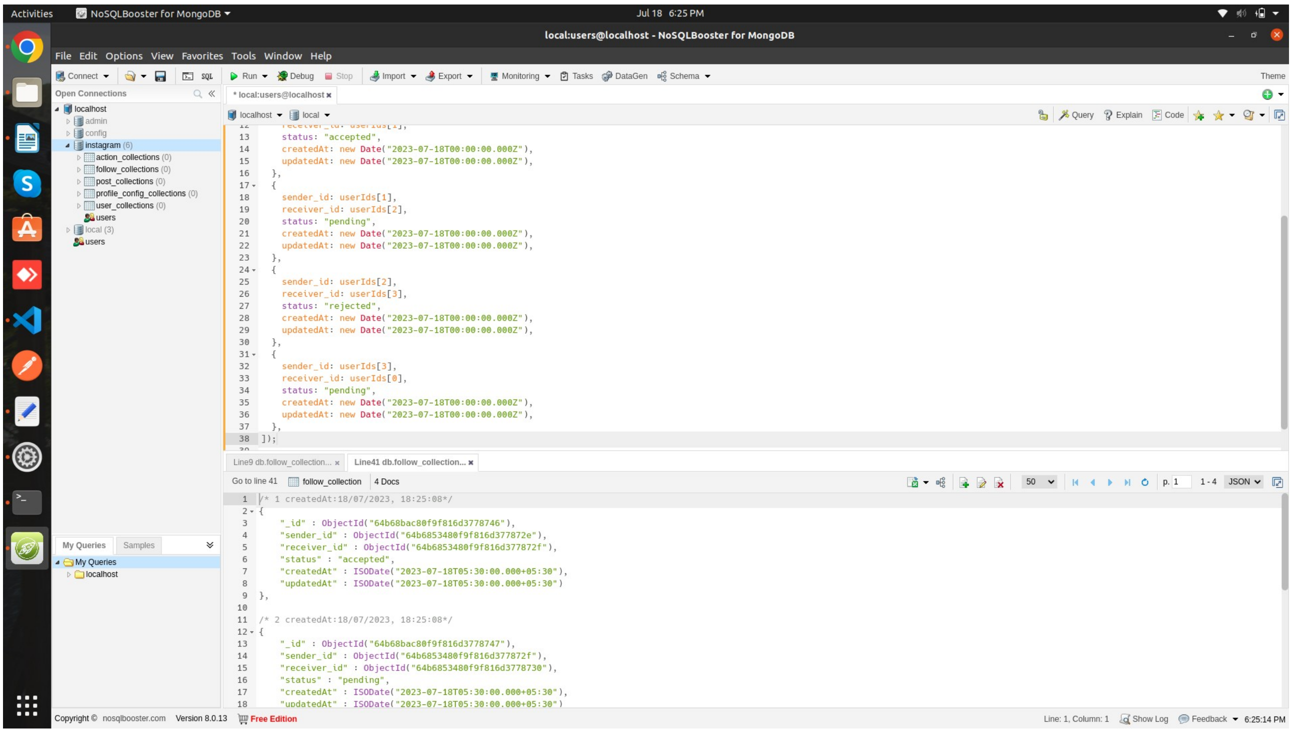
result of insertion on post_action



```
db.action_collection.find();
```

```
/* 1 createdAt:18/07/2023, 18:14:08*/
{
    "_id" : ObjectId("64b6891880f9f816d377873a"),
    "post_id" : ObjectId("64b6879a80f9f816d3778737"),
    "user_id" : ObjectId("64b6853480f9f816d377872e"),
    "action" : {
        "action_type" : "like",
        "action_content" : "Liked the post.",
        "action_date" : ISODate("2023-07-18T23:37:46.000+05:30")
    }
},

/* 2 createdAt:18/07/2023, 18:14:08*/
{
    "_id" : ObjectId("64b6891880f9f816d377873b"),
    "post_id" : ObjectId("64b6879a80f9f816d3778736"),
    "user_id" : ObjectId("64b6853480f9f816d377872f"),
    "action" : {
        "action_type" : "comment",
        "action_content" : "Nice post!",
        "action_date" : ISODate("2023-07-18T23:37:46.000+05:30")
    }
},

/* 3 createdAt:18/07/2023, 18:14:08*/
{
    "_id" : ObjectId("64b6891880f9f816d377873c"),
    "post_id" : ObjectId("64b6879a80f9f816d3778739"),
    "user_id" : ObjectId("64b6853480f9f816d3778730"),
    "action" : {
        "action_type" : "like",
        "action_content" : "Liked the post.",
        "action_date" : ISODate("2023-07-18T23:37:46.000+05:30")
    }
},

/* 4 createdAt:18/07/2023, 18:14:08*/
{
    "_id" : ObjectId("64b6891880f9f816d377873d"),
    "post_id" : ObjectId("64b6879a80f9f816d3778738"),
    "user_id" : ObjectId("64b6853480f9f816d3778731"),
    "action" : {
        "action_type" : "comment",
```

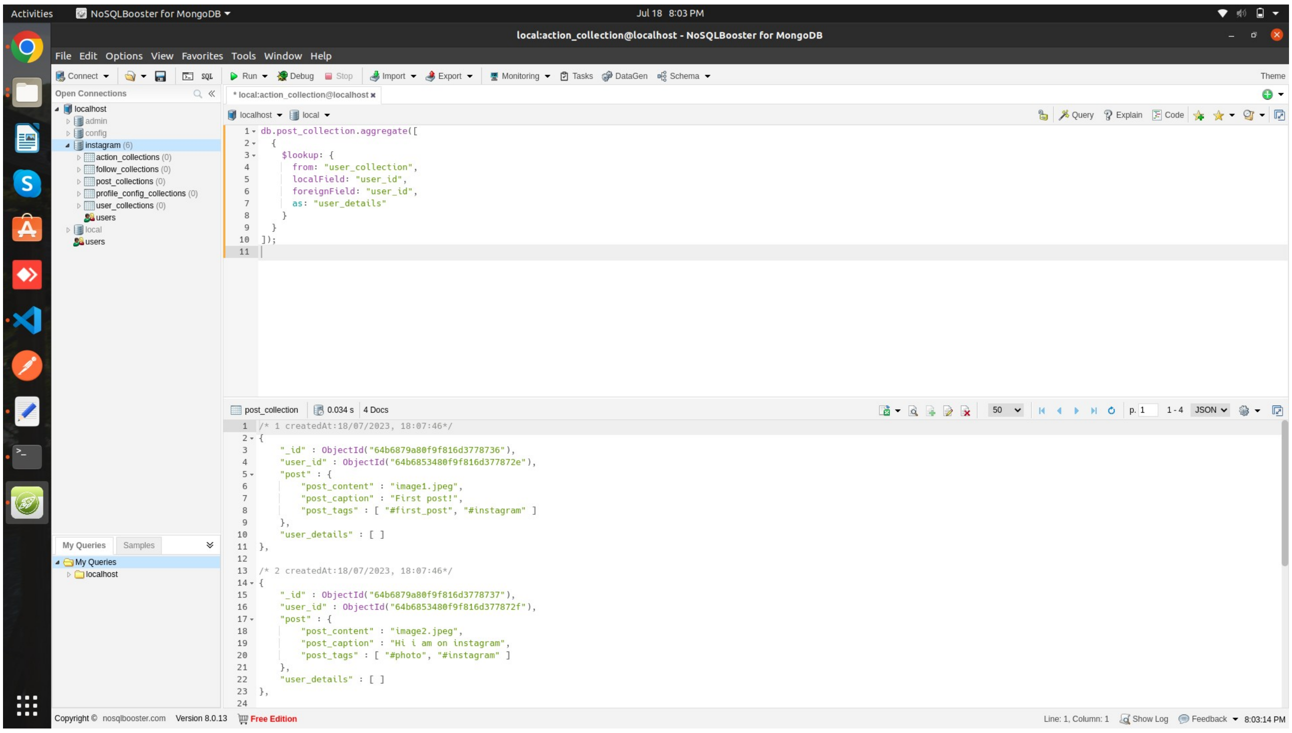
insertion on follow



```
    receiver_id: userIds[1],
    status: "accepted",
    createdAt: new Date("2023-07-18T00:00:00.000Z"),
    updatedAt: new Date("2023-07-18T00:00:00.000Z"),
  },
  {
    sender_id: userIds[1],
    receiver_id: userIds[2],
    status: "pending",
    createdAt: new Date("2023-07-18T00:00:00.000Z"),
    updatedAt: new Date("2023-07-18T00:00:00.000Z"),
  },
  {
    sender_id: userIds[2],
    receiver_id: userIds[3],
    status: "rejected",
    createdAt: new Date("2023-07-18T00:00:00.000Z"),
    updatedAt: new Date("2023-07-18T00:00:00.000Z"),
  },
  {
    sender_id: userIds[3],
    receiver_id: userIds[0],
    status: "pending",
    createdAt: new Date("2023-07-18T00:00:00.000Z"),
    updatedAt: new Date("2023-07-18T00:00:00.000Z"),
  },
]);
```

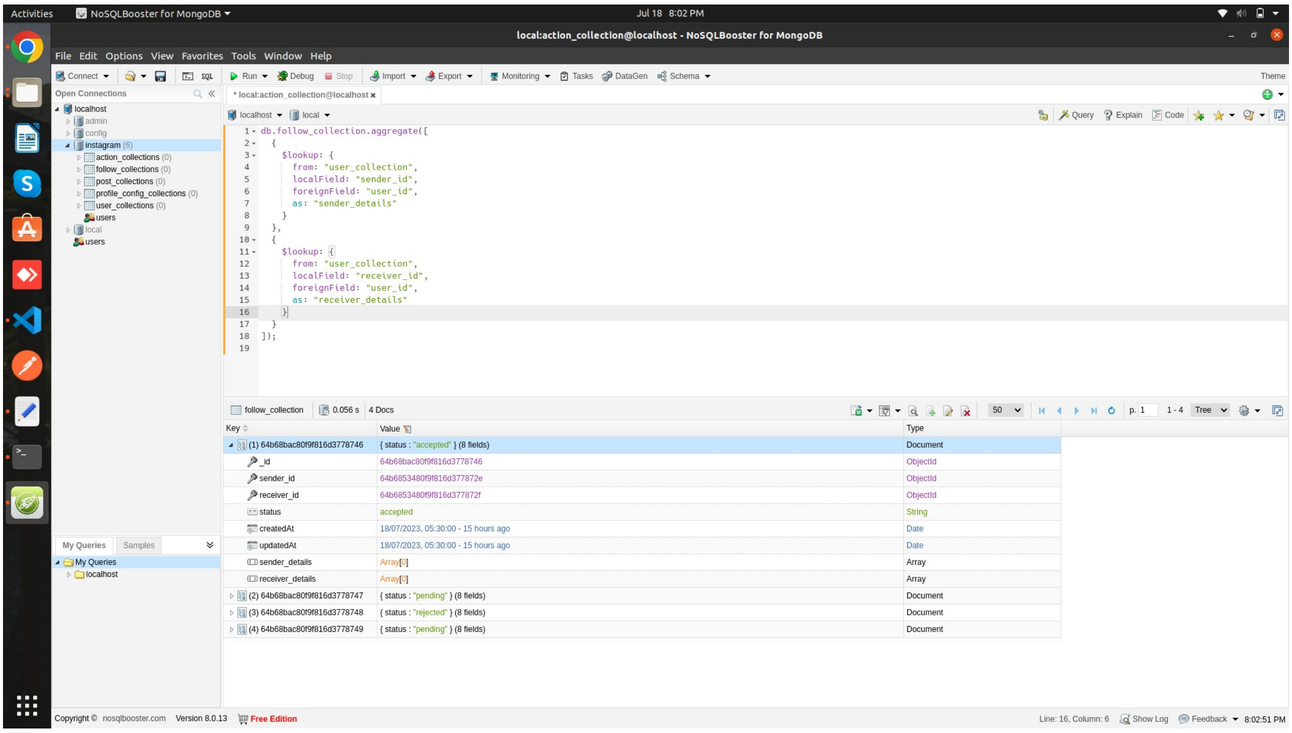
```
/* 1 createdAt:18/07/2023, 18:25:08*/
{
    "_id" : ObjectId("64b68bac80f9f816d3778746"),
    "sender_id" : ObjectId("64b6853480f9f816d377872e"),
    "receiver_id" : ObjectId("64b6853480f9f816d377872f"),
    "status" : "accepted",
    "createdAt" : ISODate("2023-07-18T05:30:00.000+05:30"),
    "updatedAt" : ISODate("2023-07-18T05:30:00.000+05:30")
},

/* 2 createdAt:18/07/2023, 18:25:08*/
{
    "_id" : ObjectId("64b68bac80f9f816d3778747"),
    "sender_id" : ObjectId("64b6853480f9f816d377872f"),
    "receiver_id" : ObjectId("64b6853480f9f816d3778730"),
    "status" : "pending",
    "createdAt" : ISODate("2023-07-18T05:30:00.000+05:30"),
    "updatedAt" : ISODate("2023-07-18T05:30:00.000+05:30")
```

Lookup queries :



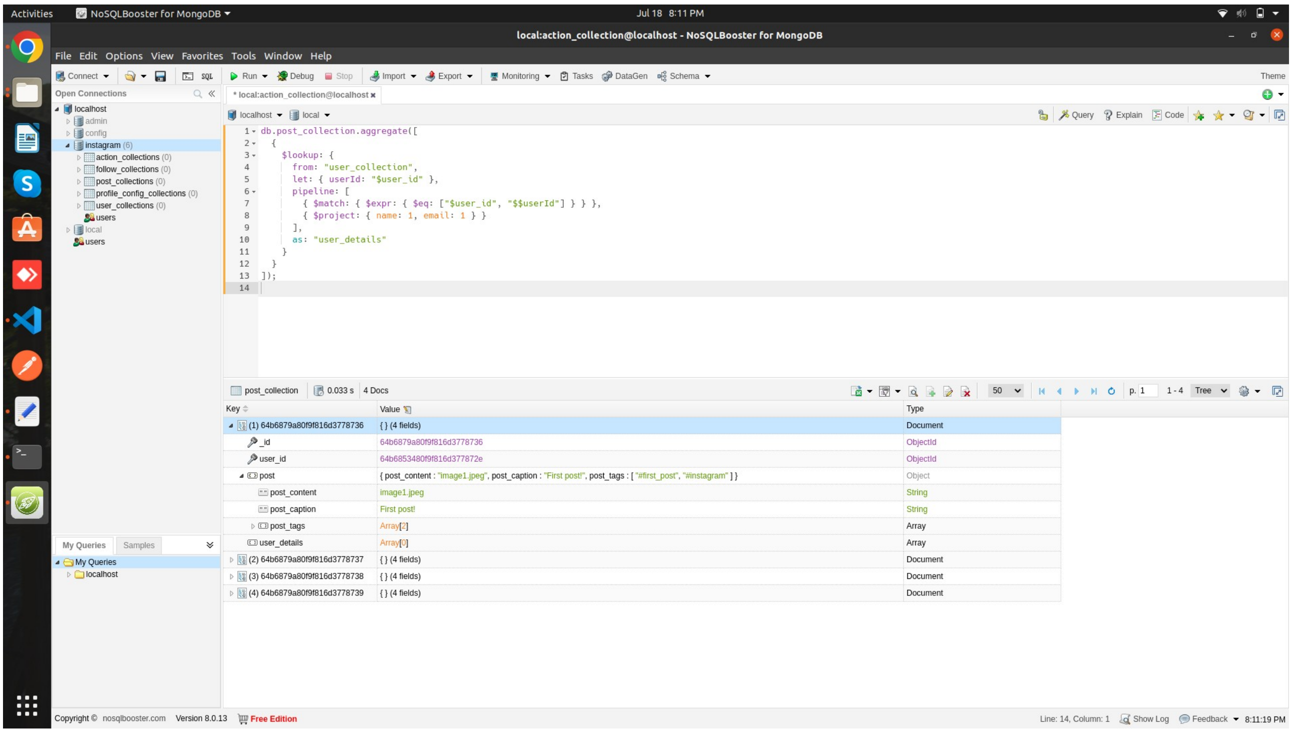
```
db.post_collection.aggregate([
  {
    $lookup: {
      from: "user_collection",
      localField: "user_id",
      foreignField: "user_id",
      as: "user_details"
    }
  }
]);
```

```
/* 1 createdAt:18/07/2023, 18:07:46*/
{
    "_id" : ObjectId("64b6879a80f9f816d3778736"),
    "user_id" : ObjectId("64b6853480f9f816d377872e"),
    "post" : {
        "post_content" : "image1.jpeg",
        "post_caption" : "First post!",
        "post_tags" : [ "#first_post", "#instagram" ]
    },
    "user_details" : [ ]
},

/* 2 createdAt:18/07/2023, 18:07:46*/
{
    "_id" : ObjectId("64b6879a80f9f816d3778737"),
    "user_id" : ObjectId("64b6853480f9f816d377872f"),
    "post" : {
        "post_content" : "image2.jpeg",
        "post_caption" : "Hi i am on instagram",
        "post_tags" : [ "#photo", "#instagram" ]
    },
    "user_details" : [ ]
},
```



```
db.follow_collection.aggregate([
  {
    $lookup: {
      from: "user_collection",
      localField: "sender_id",
      foreignField: "user_id",
      as: "sender_details"
    }
  },
  {
    $lookup: {
      from: "user_collection",
      localField: "receiver_id",
      foreignField: "user_id",
      as: "receiver_details"
    }
  }
]);
```

Key	Value	Type
(1) 64b68bac80f9f816d3778746	{ status : "accepted" } (8 fields)	Document
_id	64b68bac80f9f816d3778746	ObjectId
sender_id	64b6853480f9f816d377872e	ObjectId
receiver_id	64b6853480f9f816d377872f	ObjectId
status	accepted	String
createdAt	18/07/2023, 05:30:00 - 15 hours ago	Date
updatedAt	18/07/2023, 05:30:00 - 15 hours ago	Date
sender_details	Array[0]	Array
receiver_details	Array[0]	Array
(2) 64b68bac80f9f816d3778747	{ status : "pending" } (8 fields)	Document
(3) 64b68bac80f9f816d3778748	{ status : "rejected" } (8 fields)	Document
(4) 64b68bac80f9f816d3778749	{ status : "pending" } (8 fields)	Document

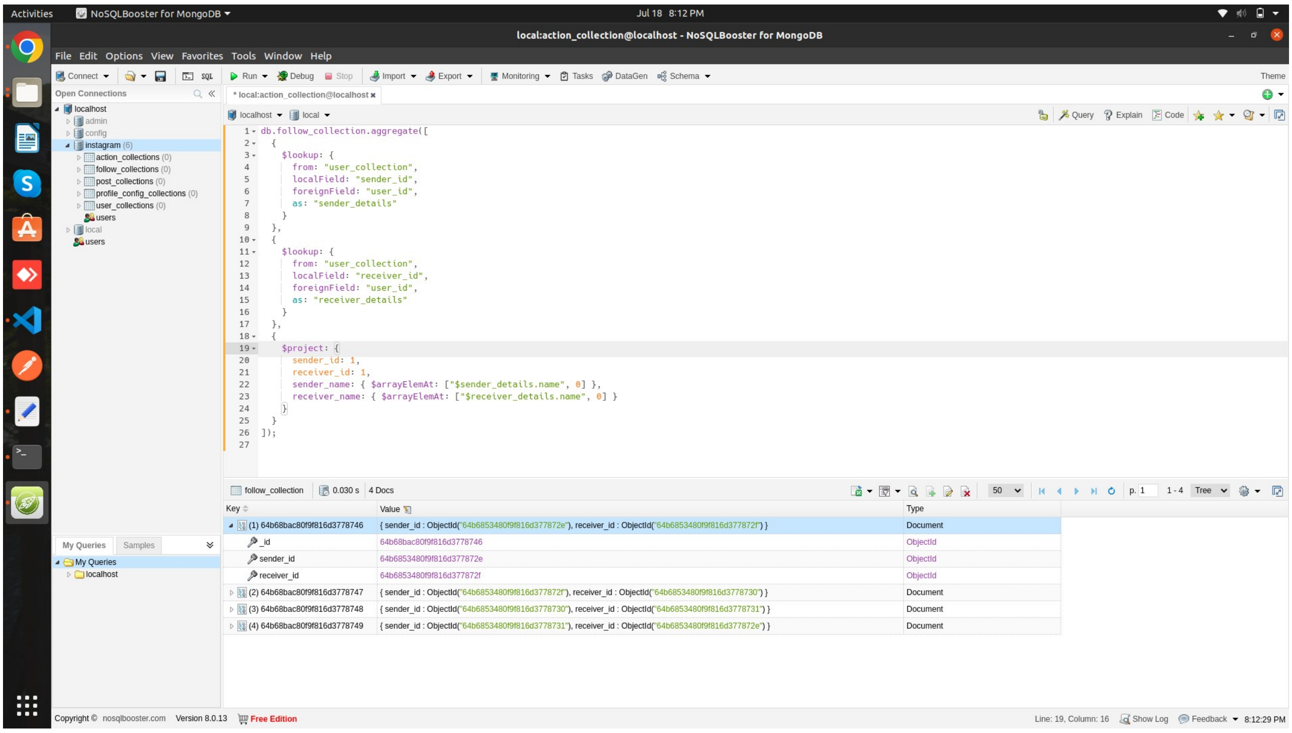


local:action_collection@localhost - NoSQLBooster for MongoDB

```
db.post_collection.aggregate([
  {
    $lookup: {
      from: "user_collection",
      let: { userId: "$user_id" },
      pipeline: [
        { $match: { $expr: { $eq: ["$user_id", "$$userId"] } } },
        { $project: { name: 1, email: 1 } }
      ],
      as: "user_details"
    }
  }
]);
```

post_collection 0.033 s 4 Docs

Key	Value	Type
(1) 64b6879a80f9f816d3778736	{ } (4 fields)	Document
_id	64b6879a80f9f816d3778736	ObjectId
user_id	64b6853480f9f816d377872e	ObjectId
post	{ post_content : "image1.jpeg", post_caption : "First post!", post_tags : ["#first_post", "#instagram"] }	Object
post_content	image1.jpeg	String
post_caption	First post!	String
post_tags	Array[2]	Array
user_details	Array[0]	Array
(2) 64b6879a80f9f816d3778737	{ } (4 fields)	Document
(3) 64b6879a80f9f816d3778738	{ } (4 fields)	Document
(4) 64b6879a80f9f816d3778739	{ } (4 fields)	Document



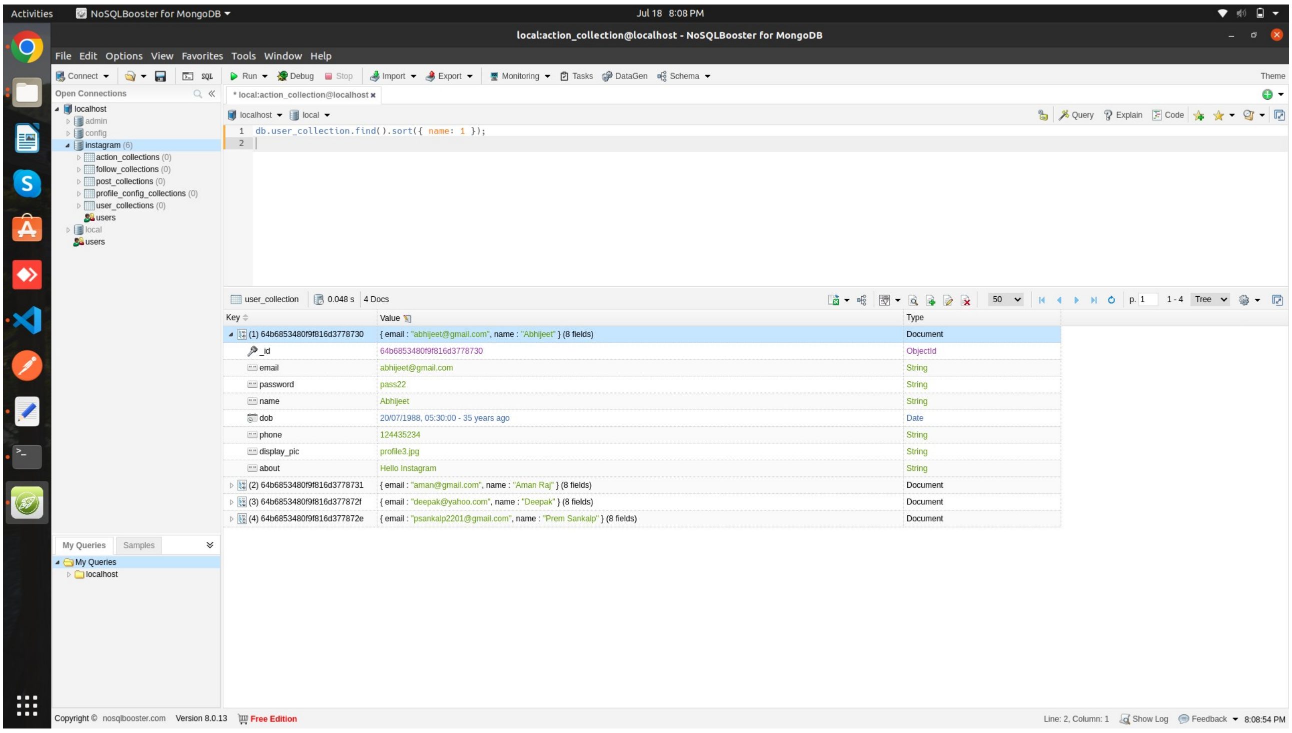
local:action_collection@localhost - NoSQLBooster for MongoDB

```
db.follow_collection.aggregate([
  {
    $lookup: {
      from: "user_collection",
      localField: "sender_id",
      foreignField: "user_id",
      as: "sender_details"
    }
  },
  {
    $lookup: {
      from: "user_collection",
      localField: "receiver_id",
      foreignField: "user_id",
      as: "receiver_details"
    }
  },
  {
    $project: {
      sender_id: 1,
      receiver_id: 1,
      sender_name: { $arrayElemAt: ["$sender_details.name", 0] },
      receiver_name: { $arrayElemAt: ["$receiver_details.name", 0] }
    }
  }
]);
```

follow_collection 0.030 s 4 Docs

Key	Value	Type
(1) 64b68bac80f9f816d3778746	{ sender_id : ObjectId("64b6853480f9f816d377872e"), receiver_id : ObjectId("64b6853480f9f816d377872f") }	Document
_id	64b68bac80f9f816d3778746	ObjectId
sender_id	64b6853480f9f816d377872e	ObjectId
receiver_id	64b6853480f9f816d377872f	ObjectId
(2) 64b68bac80f9f816d3778747	{ sender_id : ObjectId("64b6853480f9f816d377872f"), receiver_id : ObjectId("64b6853480f9f816d3778730") }	Document
(3) 64b68bac80f9f816d3778748	{ sender_id : ObjectId("64b6853480f9f816d3778730"), receiver_id : ObjectId("64b6853480f9f816d3778731") }	Document
(4) 64b68bac80f9f816d3778749	{ sender_id : ObjectId("64b6853480f9f816d3778731"), receiver_id : ObjectId("64b6853480f9f816d377872e") }	Document

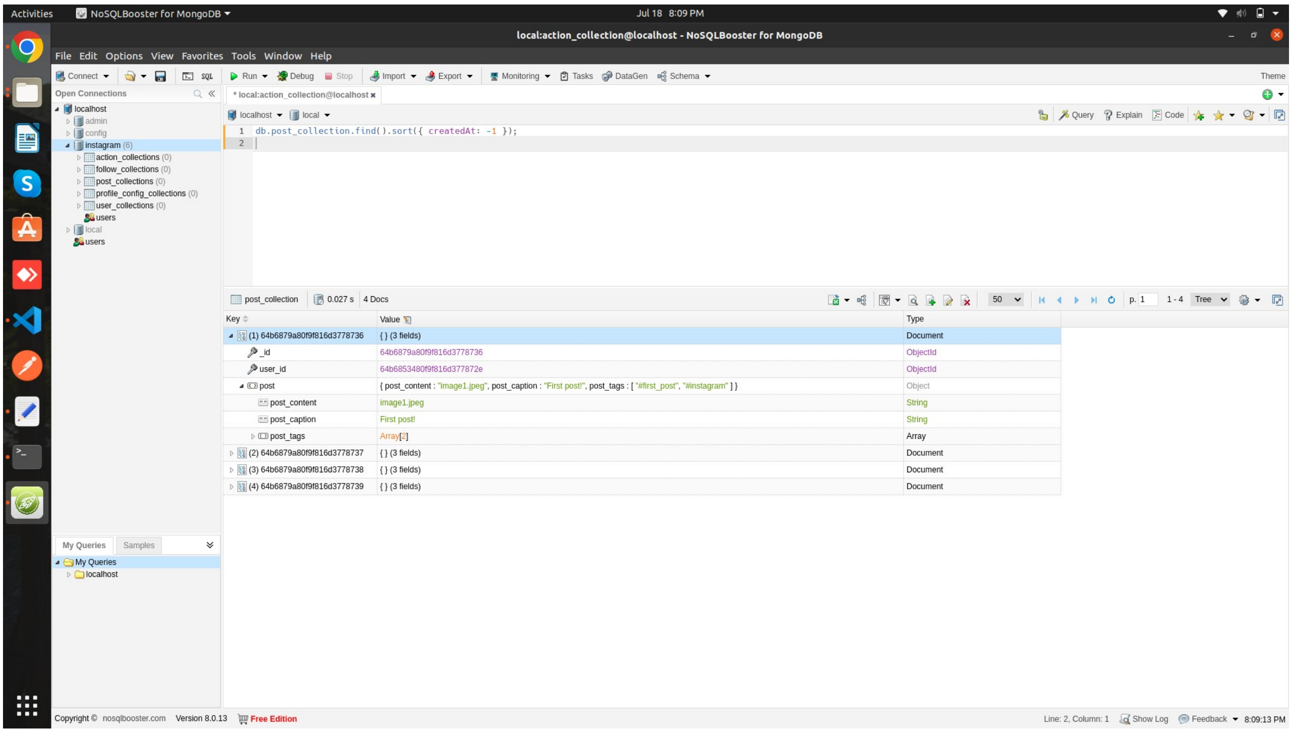
Sort queries



```
db.user_collection.find().sort({ name: 1 });
```

user_collection | 0.048 s | 4 Docs

Key	Value	Type
(1) 64b6853480f9f816d3778730	{ email : "abhijeet@gmail.com", name : "Abhijeet" } (8 fields)	Document
_id	64b6853480f9f816d3778730	ObjectId
email	abhijeet@gmail.com	String
password	pass22	String
name	Abhijeet	String
dob	20/07/1988, 05:30:00 - 35 years ago	Date
phone	124435234	String
display_pic	profile3.jpg	String
about	Hello Instagram	String
(2) 64b6853480f9f816d3778731	{ email : "aman@gmail.com", name : "Aman Raj" } (8 fields)	Document
(3) 64b6853480f9f816d377872f	{ email : "deepak@yahoo.com", name : "Deepak" } (8 fields)	Document
(4) 64b6853480f9f816d377872e	{ email : "psankalp2201@gmail.com", name : "Prem Sankalp" } (8 fields)	Document

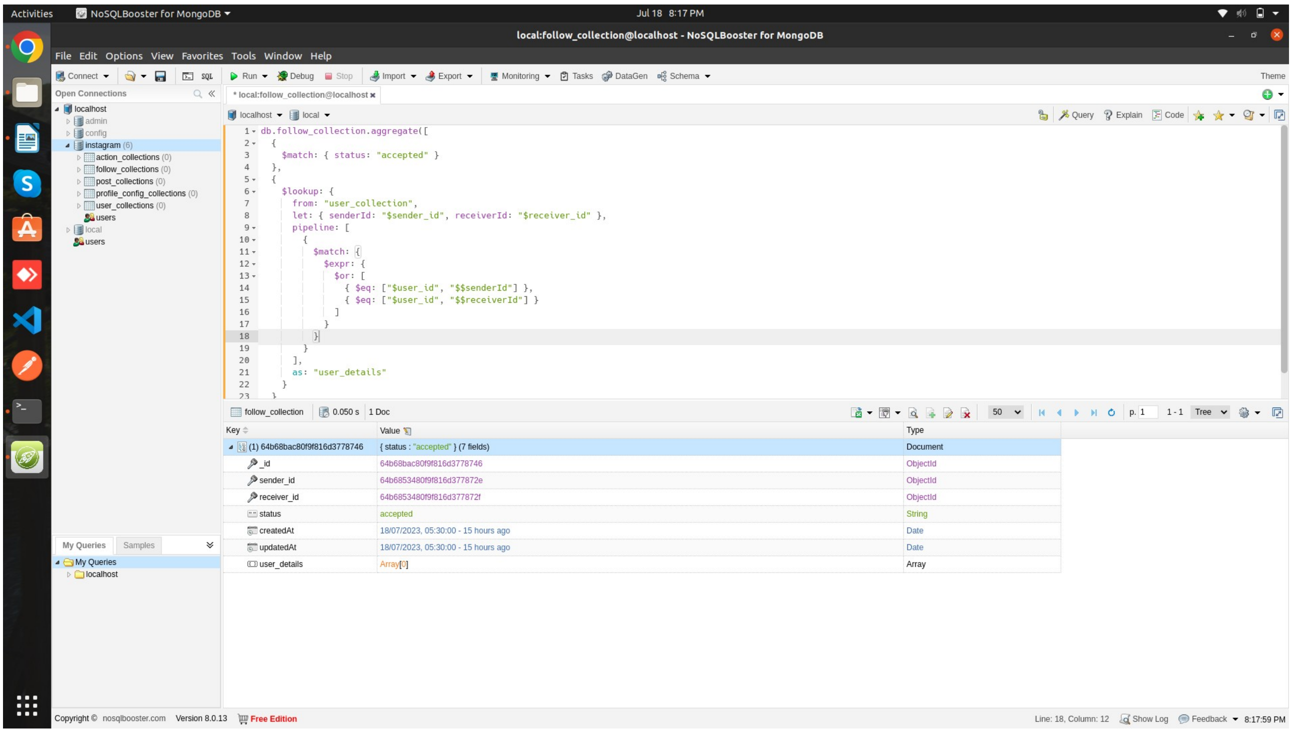


```
db.post_collection.find().sort({ createdAt: -1 });
```

post_collection | 0.027 s | 4 Docs

Key	Value	Type
(1) 64b6879a80f9f816d3778736	{ } (3 fields)	Document
_id	64b6879a80f9f816d3778736	ObjectId
user_id	64b6853480f9f816d377872e	ObjectId
post	{ post_content : "image1.jpeg", post_caption : "First post!", post_tags : ["#first_post", "#instagram"] }	Object
post_content	image1.jpeg	String
post_caption	First post!	String
post_tags	Array[2]	Array
(2) 64b6879a80f9f816d3778737	{ } (3 fields)	Document
(3) 64b6879a80f9f816d3778738	{ } (3 fields)	Document
(4) 64b6879a80f9f816d3778739	{ } (3 fields)	Document

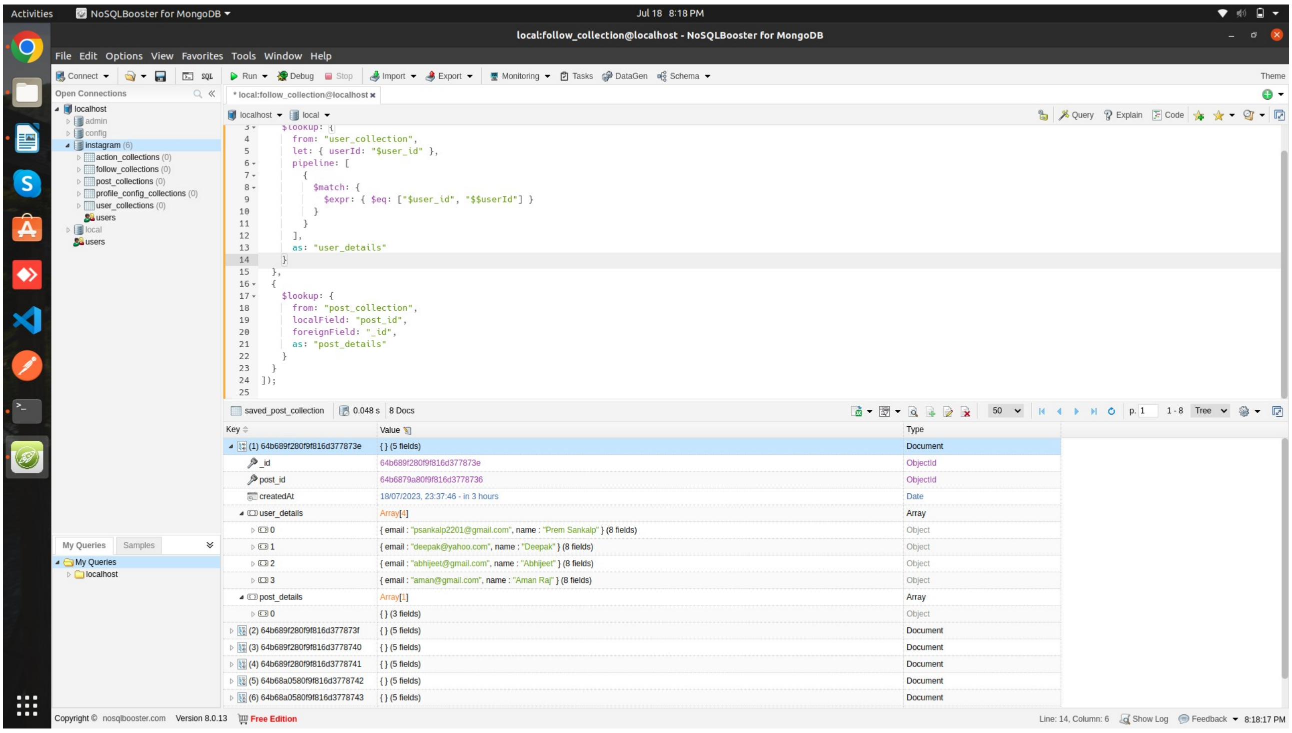
Pipeline Lookup based queries :



```
db.follow_collection.aggregate([
  {
    $match: { status: "accepted" }
  },
  {
    $lookup: {
      from: "user_collection",
      let: { senderId: "$sender_id", receiverId: "$receiver_id" },
      pipeline: [
        {
          $match: {
            $expr: {
              $or: [
                { $eq: ["$user_id", "$$senderId"] },
                { $eq: ["$user_id", "$$receiverId"] }
              ]
            }
          }
        }
      ],
      as: "user_details"
    }
  }
```

follow_collection | 0.050 s | 1 Doc | 50

Key	Value	Type
(1) 64b68bac80f9f816d3778746	{ status : "accepted" } (7 fields)	Document
_id	64b68bac80f9f816d3778746	ObjectId
sender_id	64b6853480f9f816d377872e	ObjectId
receiver_id	64b6853480f9f816d377872f	ObjectId
status	accepted	String
createdAt	18/07/2023, 05:30:00 - 15 hours ago	Date
updatedAt	18/07/2023, 05:30:00 - 15 hours ago	Date
user_details	Array[0]	Array



```
    $lookup: {
      from: "user_collection",
      let: { userId: "$user_id" },
      pipeline: [
        {
          $match: {
            $expr: { $eq: ["$user_id", "$$userId"] }
          }
        }
      ],
      as: "user_details"
    }
  },
  {
    $lookup: {
      from: "post_collection",
      localField: "post_id",
      foreignField: "_id",
      as: "post_details"
    }
  }
]);
```

saved_post_collection | 0.048 s | 8 Docs | 50

Key	Value	Type
(1) 64b689f280f9f816d377873e	{ } (5 fields)	Document
_id	64b689f280f9f816d377873e	ObjectId
post_id	64b6879a80f9f816d3778736	ObjectId
createdAt	18/07/2023, 23:37:46 - in 3 hours	Date
user_details	Array[4]	Array
0	{ email : "psankalp2201@gmail.com", name : "Prem Sankalp" } (8 fields)	Object
1	{ email : "deepak@yahoo.com", name : "Deepak" } (8 fields)	Object
2	{ email : "abhijeet@gmail.com", name : "Abhijeet" } (8 fields)	Object
3	{ email : "aman@gmail.com", name : "Aman Raj" } (8 fields)	Object
post_details	Array[1]	Array
0	{ } (3 fields)	Object
(2) 64b689f280f9f816d377873f	{ } (5 fields)	Document
(3) 64b689f280f9f816d3778740	{ } (5 fields)	Document
(4) 64b689f280f9f816d3778741	{ } (5 fields)	Document
(5) 64b68a0580f9f816d3778742	{ } (5 fields)	Document
(6) 64b68a0580f9f816d3778743	{ } (5 fields)	Document