

Network Programming

Lecture 5: Display Messages

Mahmoud El-Gayyar

elgayyar@ci.suez.edu.eg

Elgayyar.weebly.com



Message Posting Example

- 1. Create a message list query with Mongo**
- 2. Create a message list endpoint with Express*
- 3. Create a message list view in Angular*

Replace MongoDB with Mongoose -1

- *Install Mongoose*

```
../dev/back-end> npm install mongoose -save
```

- *Replace mongodb with mongoose*

```
var bodyParser=require ('body-parser');  
var mongoose= require ('mongoose') MongoClient;  
mongoose.connect("mongodb://localhost:27017/test",function (err,db){  
    if(!err){  
        console.log("we are connected to mongo");  
        database=db;  
    }  
});
```

Replace MongoDB with Mongoose -2

- *Create a schema and model*

```
//use capital letter to indicate something we can initiate
```

```
var Message= Mongoose.model('Message', {           //model Message -> collection messages
  msg:String                                       //second param: schema
});
```

- *Change app.post*

```
app.post('/api/message', function (req,res){
  console.log(req.body);
  database.collection('messages').insertOne(req.body);
  var message= new Message(req.body); //create a new message using our model
  message.save(); //just that simple
  res.status(200); //OK response
});
```

- *Try it what (don't forget to start your server)*

Server.js

Create a function to get all messages

```
//get all messages from the model
function GetMessages(){
    Message.find({}).exec(function(err, result){
        console.log(result);
    });
}

//run it once we connect to the db
mongoose.connect("mongodb://localhost:27017/test",function (err,db){
    if(!err){
        console.log("we are connected to mongo");
        GetMessages();
    }
});
```

- *Try it what (don't forget to start your server)*

Server.js

Message Posting Example

1. *Create a message list query with Mongo*
2. ***Create a message list endpoint with Express***
3. *Create a message list view in Angular*

Create a get Endpoint

```
//get all messages from the model
```

```
function GetMessages(req,res){  
    Message.find({}).exec(function(err, result){  
        res.send(result);  
    });  
}
```

```
app.get('/api/message', GetMessages); //same location 😊
```

- *Try it what (don't forget to start your server)*
 - ◆ You can try from PostMan
 - ◆ Or directly from browser as default action is get

Message Posting Example

1. *Create a message list query with Mongo*
2. *Create a message list endpoint with Express*
- 3. *Create a message list view in Angular***

Add getMessages in Controller

- *Create the function*

```
getMessages() {  
  
    this.$http.get('http://localhost:5000/api/message').then(function (result){  
        console.log(result);  
    });  
}
```

- *Call it in the constructor*

```
export class MainController {  
  
    constructor () {  
        'ngInject';  
        this.getMessages();  
    }  
  
}
```

- *Test Your App (data property)*

Main.controller.js

Set a message list

```
getMessages() {  
  
    var vm = this;  
  
    this.$http.get('http://localhost:5000/api/message').then(function (result){  
        vm.messages=result.data;  
    });  
}
```

Create a layout for the list

```
<div class="panel panel-default">
  <div class="panel-heading">Messages</div>
  <div class="panel-body">
    <ul class="list-group">
      <li class="list-group-item" ng-repeat="message in main.messages">
        {{message.msg}} <!--value of expression within the braces-->
      </li>
    </ul>
  </div>
</div>
```