

Basic Set Up

```
const express = require("express");
const app = express();
const path = require("path");
const mongoose = require("mongoose");

main()
  .then(() => {
    console.log("connection successful");
  })
  .catch((err) => console.log(err));

async function main() {
  await mongoose.connect("mongodb://127.0.0.1:27017/test");
}

app.set("views", path.join(__dirname, "views"));
app.set("view engine", "ejs");

app.get("/", (req, res) => {
  res.send("working root");
});

app.listen(8080, () => {
  console.log("app is listening");
});
```

snehagupta7385@gmail.com

```
% npm init -y
```

```
● shradhakhapra@Shradhas-MacBook-Air Mongo3 % npm i express
```

```
● shradhakhapra@Shradhas-MacBook-Air Mongo3 % npm i ejs
```

```
● shradhakhapra@Shradhas-MacBook-Air Mongo3 % npm i mongoose
```

Run `npm run audit` for details

found 0 vulnerabilities

● shradhakhapra@Shradhas-MacBook-Air Mongo3 % touch index.js

EXPLORER

...

JS index.js X

JS index.js > ...

▼ MONGO3

> node_modules

JS index.js

{ } package-lock.json

{ } package.json

```
1  const express = require("express");
2  const app = express();
3
4  app.listen(8080, () => {
5    console.log("server is listening on port 8080");
6  });
7
```

```
○ shradhakhapra@Shradhas-MacBook-Air Mongo3 % nodemon index.js
[nodemon] 3.0.1
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node index.js`
server is listening on port 8080
█
```


EXPLORER

...

JS index.js



▼ MONGO3

> node_modules

JS index.js

{ } package-lock.json

{ } package.json

JS index.js > main

```
1  const express = require("express");
2  const app = express();
3  const mongoose = require("mongoose");
4
5  main().catch(err => console.log(err));
6
7  async function main() {
8    await mongoose.connect('mongodb://127.0.0.1:27017/test');
9  }
10
11  app.get("/", (req, res) => {
12    res.send("root is working");
13  });
14
15  app.listen(8080, () => {
16    console.log("server is listening on port 8080");
17  });
```

JS index.js X

JS index.js > ...

```
1  const express = require("express");
2  const app = express();
3  const mongoose = require("mongoose");
4
5  main()
6    .then(() => {
7      console.log("connection successful");
8    })
9    .catch((err) => console.log(err));
10
11  async function main() {
12    await mongoose.connect("mongodb://127.0.0.1:27017/whatsapp");
13  }
14
15  app.get("/", (req, res) => {
16    res.send("root is working");
17  });
18
19  app.listen(8080, () => {
20    console.log("server is listening on port 8080");
21  });
22
```

EXPLORER

...

JS index.js

✕

{ } package.json

▼ MONGO3

> node_modules

> views

JS index.js

{ } package-lock.json

{ } package.json

JS index.js > ...

```
1  const express = require("express");
2  const app = express();
3  const mongoose = require("mongoose");
4  const path = require("path");
5
6  app.set("views", path.join(__dirname, "views"));
7  app.set("view engine", "ejs");
8
9  main()
10  .then(() => {
11    console.log("connection successful");
12  })
13  .catch((err) => console.log(err));
14
15  async function main() {
16    await mongoose.connect("mongodb://127.0.0.1:27017/whatsapp");
17  }
18
19  app.get("/", (req, res) => {
```

Creating the Model

Chat will have : (`_id`, `from`, `to`, `message`, `created_at`)

Creating the Model

Chat will have : (_id, from, to, message, created_at)

time,
date

website

snehagupta7385



EXPLORER

...

JS index.js

JS chat.js

▼ MONGO3

▼ models

JS chat.js

> node_modules

> views

JS index.js

{ } package-lock.json

{ } package.json

models > JS chat.js > ...

1 const mongoose = require("mongoose");

2

3



EXPLORER

...

JS index.js

JS chat.js

X

MONGO3

models

JS chat.js

> node_modules

> views

JS index.js

{ } package-lock.json

{ } package.json

models > JS chat.js > [0] chatSchema

```
1  const mongoose = require("mongoose");
2
3  const chatSchema = new mongoose.Schema({
4    from: {
5      type: String,
6      required: true,
7    },
8    to: {
9      type: String,
10     required: true,
11   },
12   msg: {
13     type: String,
14     maxLength: 50,
15   },
16   created_at: {
17     type: Date,
18     required: true,
19   },
20 });
21
```

EXPLORER

...

JS index.js

JS chat.js

×

▼ MONGO3

▼ models

JS chat.js

> node_modules

> views

JS index.js

{ } package-lock.json

{ } package.json

models > JS chat.js > ...

```
5     type: String,
6     required: true,
7   },
8   to: {
9     type: String,
10    required: true,
11  },
12  msg: {
13    type: String,
14    maxLength: 50,
15  },
16  created_at: {
17    type: Date,
18    required: true,
19  },
20 });
21
22 const Chat = mongoose.model("Chat", chatSchema);
23
24 module.exports = Chat;
25
```


EXPLORER

...

JS index.js



JS chat.js

▼ MONGO3

▼ models

JS chat.js

> node_modules

> views

JS index.js

{ } package-lock.json

{ } package.json

JS index.js > main

```
1  const express = require("express");
2  const app = express();
3  const mongoose = require("mongoose");
4  const path = require("path");
5  const Chat = require("../models/chat.js");
6
7  app.set("views", path.join(__dirname, "views"));
8  app.set("view engine", "ejs");
9
```

```
0 let chat1 = new Chat({
1   from: "neha",
2   to: "priya",
3   message: "send me your exam sheets",
4   created_at: new Date(),
5 });
6
7 chat1.save().then((res) => {
8   console.log(res);
9 });
0
```

```
chat1.save().then((res) => {  
  console.log(res);  
});
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```
Connection successful  
[nodemon] restarting due to changes...  
[nodemon] starting `node index.js`  
Server is listening on port 8080  
Connection successful  
[nodemon] restarting due to changes...  
[nodemon] starting `node index.js`  
Server is listening on port 8080  
Connection successful
```

```
from: 'neha',  
to: 'priya',  
created_at: 2023-09-16T18:27:19.505Z,  
_id: new ObjectId("6505f387de28b7b4178080ca"),  
__v: 0
```

```
amazon 112.00 KiB
config 60.00 KiB
local 72.00 KiB
test 80.00 KiB
```

```
[test> show dbs
```

```
admin 40.00 KiB
amazon 112.00 KiB
config 60.00 KiB
local 72.00 KiB
test 80.00 KiB
whatsapp 8.00 KiB
```

```
[test> use whatsapp
```

```
switched to db whatsapp
```

```
[whatsapp> show collections
```

```
chats
```

```
[whatsapp> db.chats.find()
```

```
[
```

```
{
```

```
  _id: ObjectId("6505f387de28b7b4178080ca"),
```

```
  from: 'neha',
```

```
  to: 'priya',
```

```
  created_at: ISODate("2023-09-16T18:27:19.505Z"),
```

```
  __v: 0
```

shenagupta7365@gmail.com

EXPLORER

...

▼ MONGO3

▼ models

JS chat.js

> node_modules

> views

JS index.js

{ } package-lock.json

{ } package.json

JS index.js X JS chat.js

JS index.js > [chat1] > msg

```
17 |   await mongoose.connect("mongodb://127.0.0.1:27017/whatsapp");
18 | }
19 |
20 | let chat1 = new Chat({
21 |   from: "neha",
22 |   to: "priya",
23 |   msg: "send me your exam sheets",
24 |   create mongoose
25 | });
26 |
27 | chat1.save().then((res) => {
28 |   console.log(res);
29 | });
30 |
31 | app.get("/", (req, res) => {
32 |   res.send("root is working");
33 | });
34 |
35 | app.listen(8080, () => {
36 |   console.log("server is listening on port 8080");
```

PROBLEMS

OUTPUT

TERMINAL

DEBUG CONSOLE

```
to: 'priya',
msg: 'send me your exam sheets',
created_at: 2023-09-16T18:29:53.431Z,
_id: new ObjectId("6505f4215036ec4ef81dbbbc"),
```

```
    __v: 0
  },
  {
    _id: ObjectId("6505f4215036ec4ef81dbbbc"),
    from: 'neha',
    to: 'priya',
    msg: 'send me your exam sheets',
    created_at: ISODate("2023-09-16T18:29:53.431Z"),
    __v: 0
  }
}
```

Initialize Database

init.js

EXPLORER

▼ MONGO3

▼ models

JS chat.js

> node_modules

> views

JS index.js

JS init.js

{ } package-lock.json

{ } package.json

JS index.js

JS init.js

JS init.js > ...

```
1  const mongoose = require("mongoose");
2  const Chat = require("../models/chat.js");
3
4  main()
5    .then(() => {
6      console.log("connection successful");
7    })
8    .catch((err) => console.log(err));
9
10 async function main() {
11   await mongoose.connect("mongodb://127.0.0.1:27017/whatsapp");
12 }
13
14 |
```



```
},  
{  
  from: "rohit",  
  to: "mohit",  
  message: "teach me JS callbacks",  
  created_at: new Date(),  
},  
{  
  from: "amit",  
  to: "sumit",  
  message: "all the best!",  
  created_at: new Date(),  
},  
{  
  from: "anita",  
  to: "ramesh",  
  message: "bring me some fruits",  
  created_at: new Date(),  
}
```

```
];
```

```
Chat.insertMany(allChats);
```

EXPLORER

...

JS index.js

JS init.js

MONGO3

models

JS chat.js

> node_modules

> views

JS index.js

JS init.js

{ } package-lock.json

{ } package.json

JS init.js > [0] allChhats

```
5      .then(() => {
6          console.log("connection successful");
7      })
8      .catch((err) => console.log(err));
9
10     async function main() {
11         await mongoose.connect("mongodb://127.0.0.1:27017/whatsapp");
12     }
13
14     let allChats = [
15         {
16             from: "neha",
17             to: "preeti",
18             message: "send me notes for the exam",
19             created_at: new Date(),
20         },
21         {
22             from: "rohit",
23             to: "mohit",
24             message: "teach me JS callbacks",
```

JS init.js > [x] allChats

```
25     created_at: new Date(),
26   },
27   {
28     from: "amit",
29     to: "sumit",
30     msg: "all the best!",
31     created_at: new Date(),
32   },
33   {
34     from: "anita",
35     to: "ramesh",
36     msg: "bring me some fruits",
37     created_at: new Date(),
38   },
39   {
40     from: "tony",
41     to: "peter",
42     msg: "love you 3000",
43     created_at: new Date(),
44   },
```

PROBLEMS

OUTPUT

TERMINAL

DEBUG CONSOLE

- ⊗ shradhakhapra@Shradhas-MacBook-Air Mongo3 % node init.js
connection successful
^C
- shradhakhapra@Shradhas-MacBook-Air Mongo3 % node init.js
connection successful

```
from: 'rohit',
to: 'mohit',
msg: 'teach me JS callbacks',
created_at: ISODate("2023-09-16T18:34:56.491Z"),
__v: 0
},
{
  _id: ObjectId("6505f5506e3a3c098d6c907f"),
  from: 'amit',
  to: 'sumit',
  msg: 'all the best!',
  created_at: ISODate("2023-09-16T18:34:56.491Z"),
  __v: 0
},
{
  _id: ObjectId("6505f5506e3a3c098d6c9080"),
  from: 'anita',
  to: 'ramesh',
  msg: 'bring me some fruits',
  created_at: ISODate("2023-09-16T18:34:56.491Z"),
  __v: 0
},
{
  _id: ObjectId("6505f5506e3a3c098d6c9081"),
  from: 'tony',
  to: 'peter',
```

Chats

Index Route

GET /chats

```
3 }  
9  
0 //Index Route  
1 app.get("/chats", async (req, res) => {  
2   let chats = await Chat.find();  
3   console.log(chats);  
4   res.send("working");  
5 });  
6
```



localhost:8080/chats

working


```
},  
{  
  _id: new ObjectId("6505f5506e3a3c098d6c9080"),  
  from: 'anita',  
  to: 'ramesh',  
  msg: 'bring me some fruits',  
  created_at: 2023-09-16T18:34:56.491Z,  
  __v: 0  
},  
{  
  _id: new ObjectId("6505f5506e3a3c098d6c9081"),  
  from: 'tony',  
  to: 'peter',  
  msg: 'love you 3000',  
  created_at: 2023-09-16T18:34:56.491Z,  
  __v: 0  
}
```

1

//Index Route

```
app.get("/chats", async (req, res) => {  
  let chats = await Chat.find();  
  console.log(chats);  
  res.render("index.ejs", {chats});  
});
```

EXPLORER

...

JS index.js

<> index.ejs •

✓ MONGO3

> models

> node_modules

✓ views

<> index.ejs

JS index.js

JS init.js

{ } package-lock.json

{ } package.json

views > <> index.ejs > div > p > ?

```
4     <meta charset="UTF-8" />
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <title>Mini Whatsapp</title>
7 </head>
8 <body>
9     <h3>All Chats!</h3>
10
11 </body>
12 </html>
13
14 for(let chat of chats) {
15     <div>
16         <p>From : <%= chat.from %> </p>
17         <p><%= chat.msg %> </p>
18         <p>Received by <%= chat.to %> </p>
19     </div>
20 }
21
```

EXPLORER

...

JS index.js

<> index.ejs ●

▼ MONGO3

> models

> node_modules

▼ views

<> index.ejs

JS index.js

JS init.js

{ } package-lock.json

{ } package.json

views > <> index.ejs > div.chat > div.msg

```
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width,
6      <title>Mini Whatsapp</title>
7    </head>
8    <body>
9      <h3>All Chats!</h3>
10
11    </body>
12  </html>
13
14  for(let chat of chats) {
15    <div class="chat">
16      <p>From : <%= chat.from %> </p>
17      <div class="msg">
18        <p><%= chat.msg %> </p>
19      </div>
20      <p>Received by <%= chat.to %> </p>
21    </div>
22  }
```

EXPLORER

...

JS index.js

<> index.ejs ●

▼ MONGO3

> models

> node_modules

▼ views

<> index.ejs

JS index.js

JS init.js

{ } package-lock.json

{ } package.json

views > <> index.ejs > html > body > ? > ?

<meta name= viewport content= width=device-width, initial-scale=1>

6 <title>Mini Whatsapp</title>

7 </head>

8 <body>

9 <h3>All Chats!</h3>

10 <% for(let chat of chats) { %>

11 <div class="chat">

12 <p>From : <%= chat.from %> </p>

13 <div class="msg">

14 <p><%= chat.msg %> </p>

15 </div>

16 <p>Received by <%= chat.to %> </p>

17 </div>

18 <% } %>

19 </body>

20 </html>

21

22

All Chats!

From : neha

send me notes for the exam

Received by preeti

From : rohit

teach me JS callbacks

Received by mohit

From : amit

all the best!

Received by sumit

From : anita

bring me some fruits

Received by ramesh

From : tony

love you 3000

Received by peter

EXPLORER

...

JS index.js X # style.css <> index.ejs

JS index.js > ...

▼ MONGO3

> models

> node_modules

▼ public

style.css

▼ views

<> index.ejs

JS index.js

JS init.js

{ } package-lock.json

{ } package.json

```
1  const express = require("express");
2  const app = express();
3  const mongoose = require("mongoose");
4  const path = require("path");
5  const Chat = require("../models/chat.js");
6
7  app.set("views", path.join(__dirname, "views"));
8  app.set("view engine", "ejs");
9  app.use(express.static(path.join(__dirname, "public")));
10
11  main()
12    .then(() => {
13      console.log("connection successful");
14    })
15    .catch((err) => console.log(err));
16
17  async function main() {
18    await mongoose.connect("mongodb://127.0.0.1:27017/whatsapp");
19  }
```

JS index.js

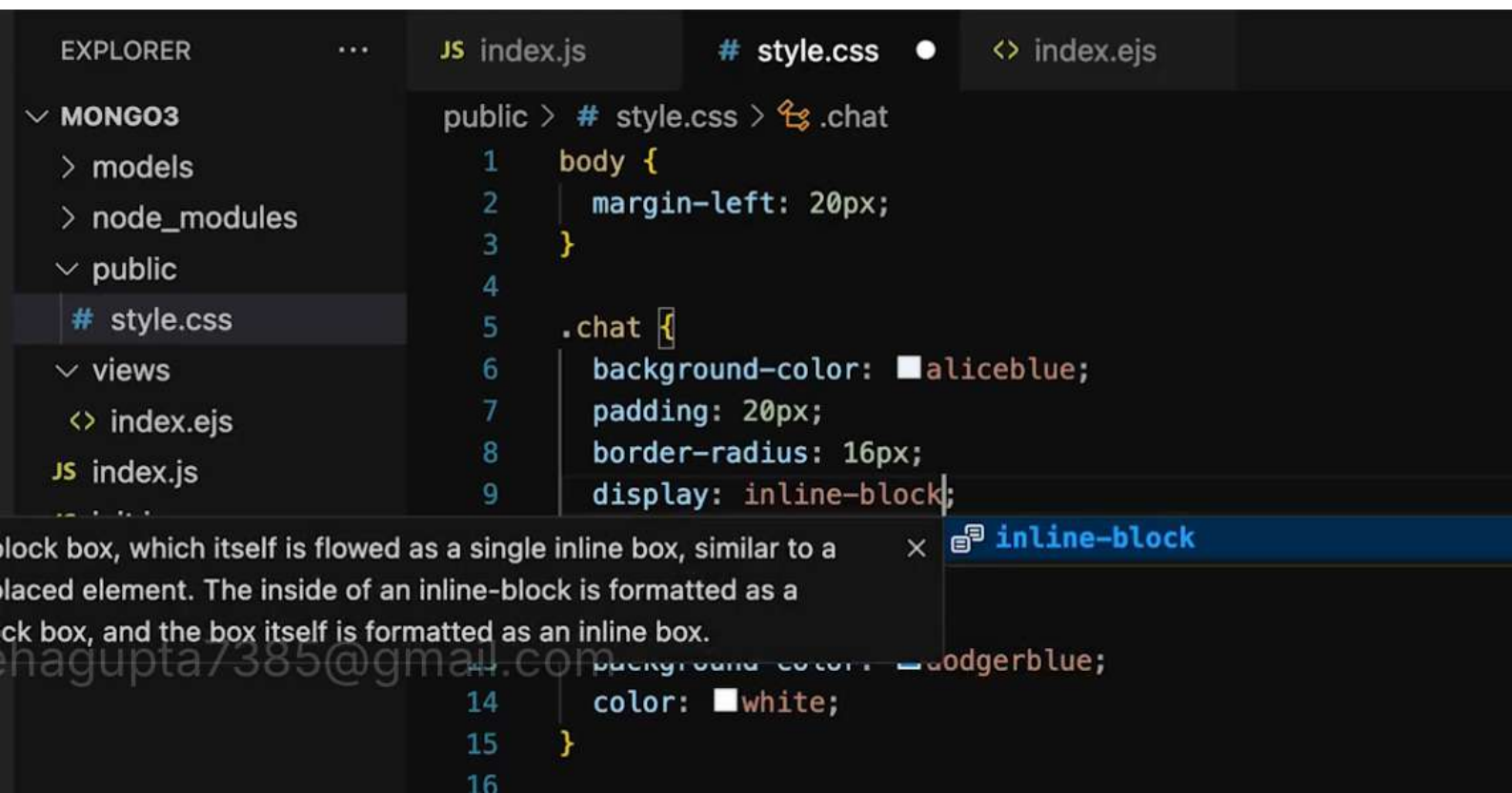
style.css ×

<> index.ejs

public > # style.css > .msg

```
1  .chat {  
2    background-color: ■aliceblue;  
3  }  
4  
5  .msg {  
6    background-color: ■dodgerblue;  
7  }  
8
```





All Chats!

From : neha

send me notes for the exam

Received by preeti

From : rohit

teach me JS callbacks

Received by mohit

From : amit

all the best!

Received by sumit

From : anita

bring me some fruits

Received by ramesh

From : tony

love you 3000

Received by peter

EXPLORER

...

JS index.js

style.css X

<> index.ejs

MONGO3

> models

> node_modules

public

style.css

views

<> index.ejs

JS index.js

JS init.js

{ } package-lock.json

{ } package.json

public > # style.css > .msg

```
1  body {
2    margin-left: 20px;
3  }
4
5  .chat {
6    background-color: ■aliceblue;
7    padding: 20px;
8    border-radius: 16px;
9    display: inline-block;
10 }
11
12 .msg {
13   background-color: ■dodgerblue;
14   color: ■white;
15   padding: 5px 15px;
16   border-radius: 16px;
17 }
18
```

EXPLORER

...

JS index.js# style.css<> index.ejs X

▼ MONGO3

> models

> node_modules

▼ public

style.css

▼ views

<> index.ejs

JS index.js

JS init.js

{ } package-lock.json

{ } package.json

views > <> index.ejs > html > body > ? > div.chat > p > i > ?

5<meta name="viewport" content="width=device-width, initial-scale=1.0" />

6<title>Mini Whatsapp</title>

7<link rel="stylesheet" href="style.css" />

8</head>

9<body>

10<h3>All Chats!</h3>

11<% for(let chat of chats) { %>

12<div class="chat">

13<p>From : <i><%= chat.from %></i></p>

14<div class="msg">

15<p><%= chat.msg %></p>

16</div>

17<p>Received by<i><%= chat.to %></i></p>

18<hr />

19</div>

20

21<% } %>

22</body>

23</html>

24

All Chats!

From : *neha*

send me notes for the exam

Received by *preeti*

From : *rohit*

teach me JS callbacks

Received by *mohit*

From : *amit*

all the best!

Received by *sumit*

Chats

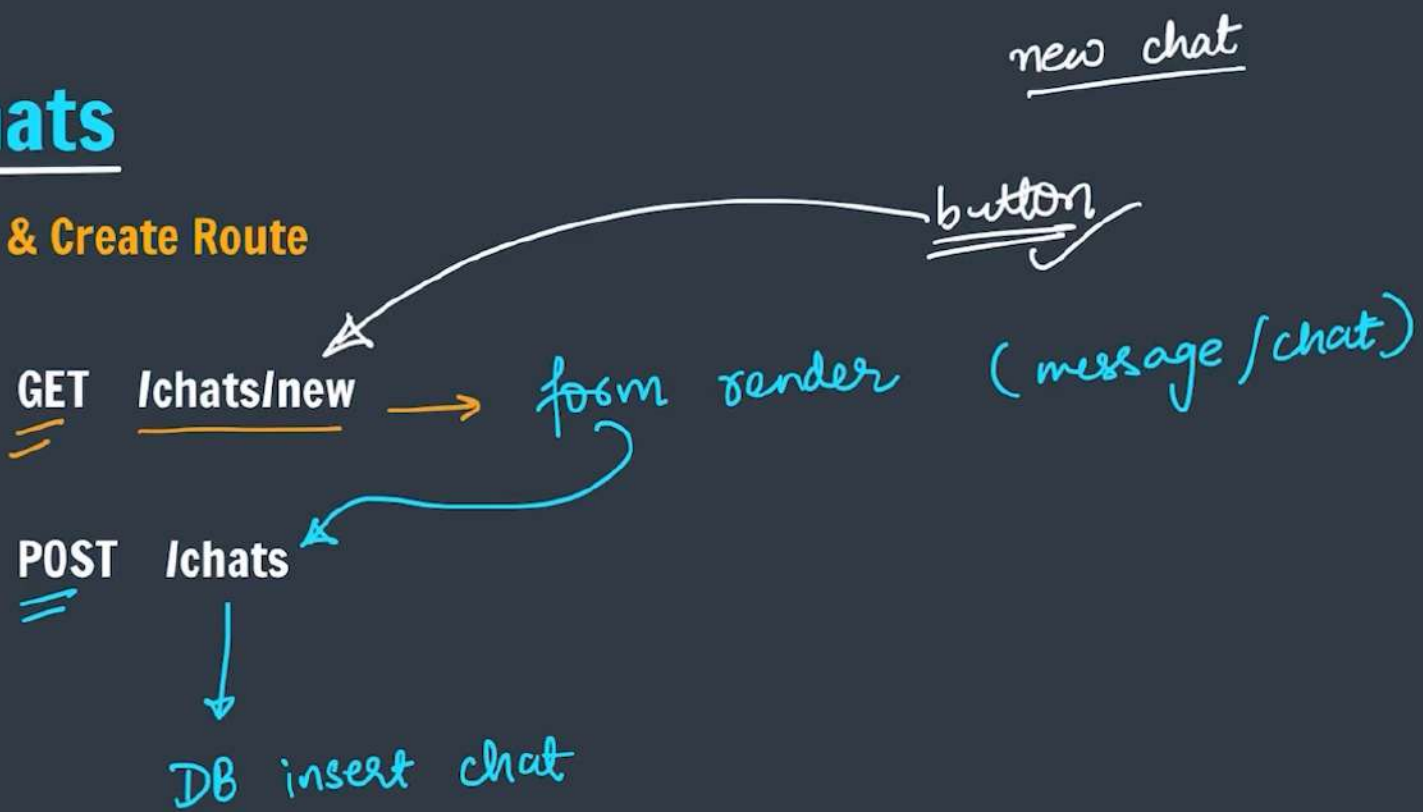
New & Create Route

GET /chats/new

POST /chats

Chats

New & Create Route



```
<body>
```

```
  <h3>All Chats!</h3>
```

```
  <form>
```

```
    <button>New Chat</button>
```

```
  </form>
```

```
<br />
```



```
8 </head>
9 <body>
10   <h3>All Chats!</h3>
11   <form method="GET" action="/chats/new">
12     | <button>New Chat</button>
13   </form>
14   <br />
```

abc New

All Chats!

New Chat



EXPLORER

...

JS index.js

<> new.ejs

▼ MONGO3

> models

> node_modules

> public

▼ views

<> index.ejs

<> new.ejs

JS index.js

JS init.js

{ } package-lock.json

{ } package.json

views > <> new.ejs > html > body > form > button

1 <!DOCTYPE html>

2 <html lang="en">

3 <head>

4 <meta charset="UTF-8" />

5 <meta name="viewport" content="width=device-width, initial-scale=1.0" />

6 <title>Write new chat</title>

7 </head>

8 <body>

9 <h3>Write new chat.</h3>

10 <form>

11 <input placeholder="who are you?" name="from" />

12 <textarea name="msg">Write your message here.</textarea>

13 <input placeholder="add receiver" name="to" />

14 <button>Create Chat</button>

15 </form>

16 </body>

17 </html>

18

abc chat

abc charset

snehagu

```
26     });  
27  
28     //New Route  
29     app.get("/chats/new", (req, res) => {  
30         res.render("new.ejs");  
31     });  
32
```

EXPLORER

...

JS index.js

<> new.ejs

X

MONGO3

> models

> node_modules

> public

v views

<> index.ejs

<> new.ejs

JS index.js

JS init.js

{ } package-lock.json


{ } package.json

views > <> new.ejs > html > body > form > br

```
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8" />
5   <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6   <title>Write new chat</title>
7 </head>
8 <body>
9   <h3>Write new chat..</h3>
10  <form>
11    <input placeholder="who are you?" name="from" />
12    <br /><br />
13    <textarea name="msg">Write your message here..</textarea>
14    <br /><br />
15    <input placeholder="add receiver" name="to" />
16    <br /><br />
17    <button>Create Chat</button>
18  </form>
19 </body>
20 </html>
21
```

Write new chat..

who are you? 

Write your message
here.. 

add receiver

Create Chat

EXPLORER

...

JS index.js

<> new.ejs

●

MONGO3

> models

> node_modules

> public

v views

<> index.ejs

<> new.ejs

JS index.js

JS init.js

{ } package-lock.json

{ } package.json

views > <> new.ejs > html > body > form

2 <html lang="en">

3 <head>

4 <meta charset="UTF-8" />

5 <meta name="viewport" content="width=device-width, initial-scale=1.0" />

6 <title>Write new chat</title>

7 </head>

8 <body>

9 <h3>Write new chat.</h3>

10 <form method="POST" action="/chats">

11 <input placeholder="who are you?" name="from" />

12

13 <textarea name="msg">Write your message here.</textarea>

14

15 <input placeholder="add receiver" name="to" />

16

17 <button>Create Chat</button>

18 </form>

19 </body>

20 </html>

```
app.set("views", path.join(__dirname, "views"));
app.set("view engine", "ejs");
app.use(express.static(path.join(__dirname, "public")));
app.use(express.urlencoded{extended: true})
```


4 //Create Route

```
5 app.post("/chats", (req, res) => {  
6   let { from, to, msg } = req.body;  
7   let newChat = new Chat({  
8     from: from,  
9     to: to,  
10    msg: msg,  
11    created_at: new Date()  
12  });  
13 });
```

```
32   });
33
34   ⚡ Create Route
35   app.post("/chats", (req, res) => {
36     let { from, to, msg } = req.body;
37     let newChat = new Chat({
38       from: from,
39       to: to,
40       msg: msg,
41       created_at: new Date(),
42     });
43     console.log(newChat);
44     res.send("working");
45   });
46
47   app.get("/", (req, res) => {
48     res.send("root is working");
49   });
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

Write new chat..

shradha

share your notes

shreay

Create Chat

0



localhost:8080/chats

working

```
from: 'shradha',  
to: 'shreya',  
msg: 'share your notes',  
created_at: 2023-09-16T19:00:56.472Z,  
_id: new ObjectId("6505fb68f5b719ccbc01e22b")
```

Ln 33, Col 1 Spaces: 4 UT

newChat

```
.save()  
.then((res) => {  
  console.log("chat was saved");  
})  
.catch((err) => {  
  console.log(err);  
});
```

```
res.redirect("/chats");  
});
```

Write new chat..

shradha

let's study web dev

shreya

Create Chat

Received by *sumit*

From : *anita*

bring me some fruits

Received by *ramesh*

From : *tony*

love you 3000

Received by *peter*

From : *shradha*

let's study web dev

Received by *shreya*

```
}  
]  
^C  
shradhakhapra@Shradhas-MacBook-Air Mongo3 %
```



This site can't be reached

localhost refused to connect.

Try:

- Checking the connection
- **Checking the proxy and the firewall**

ERR_CONNECTION_REFUSED



Details

Reload

Received by *sumit*

From : *anita*

bring me some fruits

Received by *ramesh*

From : *tony*

love you 3000

Received by *peter*

From : *shradha*

let's study web dev

Received by *shreya*

Using Date

```
<%= chat.created_at.toString().split(" ")[4] %> <br />  
<%= chat.created_at.toString().split(" ").slice(0, 4).join(" ") %> <br />
```

```
    <p><%= chat.msg %></p>
</div>
<p>Received by<i> <%= chat.to %></i></p>
<hr />
<p><%= chat.created_at.toString().split(" ") %></p>
</div>
<br /><br />
```

Received by prech

Sun, Sep, 17, 2023, 00:04:56, GMT+0530, (India, Standard, Time)

```
<p>Received by<i> <%= chat.to %></i></p>
<hr />
<p><%= chat.created_at.toString().split(" ")[4] %></p>
</div>
<br /><br />
```

Received by *preeti*

00:04:56



From : *rohit*

teach me JS callbacks

Received by *mohit*

00:04:56


```
</div>
<p>Received by<i> <%= chat.to %></i></p>
<hr />
<p><%= chat.created_at.toString().split(" ")[4] %></p>
<p><%= chat.created_at.toString().split(" ").slice(0, 4) %></p>
</div>
<br /><br />
<% } %>
```

From : *neha*

send me notes for the exam

Received by *preeti*

00:04:56

Sun, Sep, 17, 2023

From : *rohit*

teach me JS callbacks

Received by *mohit*

00:04:56

Sun, Sep, 17, 2023

nenagupta7385@gmail.com

```
<p>received by</p><p>chat.to %</p></div>  
<hr />  
<p><%= chat.created_at.to_string().split(" ")[4] %></p>  
<p><%= chat.created_at.to_string().split(" ").slice(0, 4).join("") %></p>  
</div>  
<br /><br />
```

New Chat

From : *neha*

send me notes for the exam

Received by *preeti*

00:04:56

Sun, Sep, 17, 2023

JS index.js

index.ejs ●

```
views > <> index.ejs > html > body > ? > div.chat > p > ?
```

```

9 <body>
10 <h3>All Chats!</h3>
11 <form method="GET" action="/chats/new">
12 | <button>New Chat</button>
13 </form>
14 <br />
15 <% for(let chat of chats) { %>
16 <div class="chat">
17 | <p>From : <i><%= chat.from %></i></p>
18 | <div class="msg">
19 | | <p><%= chat.msg %></p>
20 | | </div>
21 | <p>Received by<i> <%= chat.to %></i></p>
22 | <hr />
23 | <p><%= chat.created_at.toString().split(" ")[4] %></p>
24 | <p><%= chat.created_at.toString().split(" ").slice(0, 4).join("<div>
25 | </div>
26 | <br /><br />
27 | <% } %>
28 </body>
29 </html>

```

From : *neha*

send me notes for the exam

Received by *preeti*

00:04:56

☞ Sun Sep 17 2023

Chats

Edit & Update Route

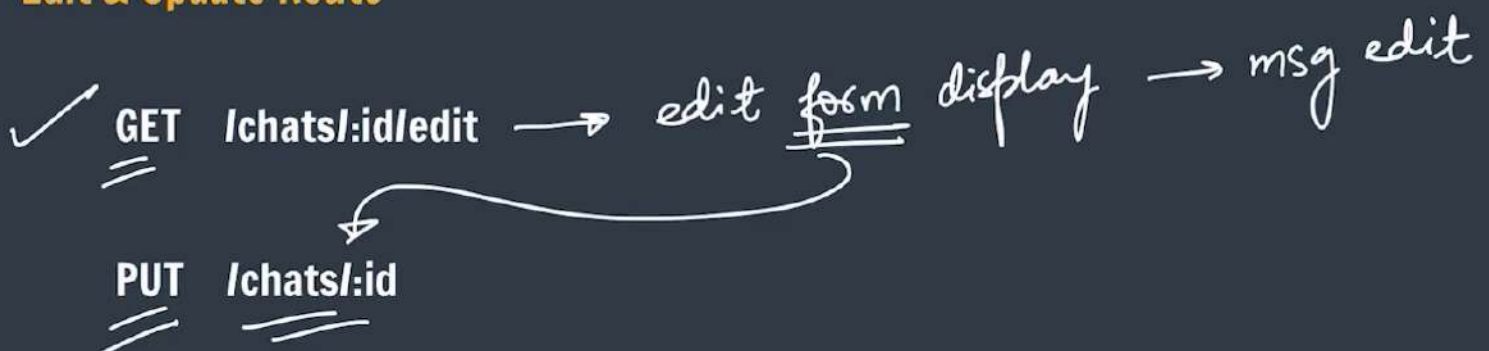
GET /chats/:id/edit

PUT /chats/:id

Chats

Edit & Update Route

- 1) edit
- 2) Edit



JS index.js

<> index.ejs

views > <> index.ejs > html > body > ? > div.chat > div.msg > form > button

```
12     <button>New Chat</button>
13   </form>
14   <br />
15   <% for(let chat of chats) { %>
16     <div class="chat">
17       <p>From : <i><%= chat.from %></i></p>
18       <div class="msg">
19         <p><%= chat.msg %></p>
20         <form>
21           <button>edit</button>
22         </form>
23       </div>
24       <p>Received by<i> <%= chat.to %></i></p>
25       <hr />
26       <p><%= chat.created_at.toString().split(" ")[4] %></p>
27       <p><%= chat.created_at.toString().split(" ").slice(0, 4).join("-") %></p>
28     </div>
29     <br /><br />
30   <% } %>
31 </body>
32 </html>
```

All Chats!

New Chat

From : *neha*

send me notes for the exam

edit

Received by *preeti*

00:04:56

Sun-Sep-17-2023

From : *rohit*

teach me JS callbacks

edit

Received by *mohit*

00:04:56

Sun-Sep-17-2023

JS index.js × <> index.ejs ×

views > <> index.ejs > html > body > ? > div.chat > div.msg > form

```
12     <button>New Chat</button>
13 </form>
14 <br />
15 <% for(let chat of chats) { %>
16   <div class="chat">
17     <p>From : <i><%= chat.from %></i></p>
18     <div class="msg">
19       <p><%= chat.msg %></p>
20       <form method="GET" action="/chats/<%= chat._id %>/edit">
21         <button>edit</button>
22       </form>
23     </div>
24     <p>Received by<i> <%= chat.to %></i></p>
25     <hr />
26     <p><%= chat.created_at.toString().split(" ")[4] %></p>
27     <p><%= chat.created_at.toString().split(" ").slice(0, 4).join("-") %>
28   </div>
29   <br /><br />
30 <% } %>
31 </body>
32 </html>
```

```
//Edit Route
```

```
app.get("/chats/:id/edit", (req, res) => {  
  res.render("edit.ejs");  
})
```

```
5 //Edit Route
```

```
7 app.get("/chats/:id/edit", async (req, res) => {  
8   let { id } = req.params;  
9   let chat = await Chat.findById(id);  
10  ⚡ res.render("edit.ejs", { chat });  
11  });
```

JS index.js

<> edit.ejs

×

<> index.ejs

views > <> edit.ejs > html > body > form > button

```
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6      <title>Edit your chat</title>
7    </head>
8    <body>
9      <h3>Edit your chat: <%= chat._id %></h3>
10     <p>This chat was sent from <%= chat.from %> to <%= chat.to %></p>
11
12     <form>
13       <textarea name="msg" rows="5" cols="10"><%= chat.msg %></textarea>
14       <button>Edit</button>
15     </form>
16   </body>
17 </html>
18
```

Edit your chat: 6505f5506e3a3c098d6c907d



This chat was sent from neha to preeti

send me
notes for
the exam

Edit

```
<title>Edit your chat</title>
</head>
<body>
  <p>Edit your chat: <%= chat._id %></p>
  <p>
    This chat was sent from <b><%= chat.from %></b> to <b><%= chat.to%></b>
  </p>

  <form>
    <textarea name="msg" rows="5" cols="10"><%= chat.msg %></textarea>
    <button>Edit</button>
  </form>
</body>
</html>
```


Edit your chat: 6505f5506e3a3c098d6c907d

This chat was sent from **neha** to **preeti**

send me
notes for
the exam

Edit

EXPLORER

...

JS index.js

<> edit.ejs

✕

<> index.ejs

MONGODB

> models

> node_modules

> public

✓ views

<> edit.ejs

<> index.ejs

<> new.ejs

JS index.js

JS init.js

{ } package-lock.json

{ } package.json

views > <> edit.ejs > html > body > form > textarea

1 <!DOCTYPE html>

2 <html lang="en">

3 <head>

4 <meta charset="UTF-8" />

5 <meta name="viewport" content="width=device-width, initial-scale=1.0" />

6 <title>Edit your chat</title>

7 </head>

8 <body>

9 <p>Edit your chat: <%= chat._id %></p>

10 <p>

11 | This chat was sent from <%= chat.from %> to <%= chat.to%>

12 </p>

13

14 <form>

15 | <textarea name="msg" rows="6" cols="50"><%= chat.msg %></textarea>

16 |
|

17 | <button>Edit</button>

18 </form>

19 </body>

20 </html>

21

Edit your chat: 6505f5506e3a3c098d6c907d

This chat was sent from **neha** to **preeti**

send me notes for the practical exam

Edit


```
]
^C%
● shradhakhapra@Shradhas-MacBook-Air Mongo3 % npm i method-override

added 2 packages, and audited 101 packages in 2s

11 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

```
<form method="POST" action="/chats/<%= chat._id%>?_method=PUT">
  <textarea name="msg" rows="6" cols="50"><%= chat.msg %></textarea>
  <br />
  <button>Edit</button>
</form>
</body>
```

index.js > app.put(/chats/:id) callback

```
62
63 //Update Route
64 app.put("/chats/:id", async (req, res) => {
65     let { id } = req.params;
66     let { newMsg } = req.body;
67     let updatedChat = await Chat.findByIdAndUpdate(
68         id,
69         { msg: newMsg },
70         { runValidators: true, new: true }
71     );
72
73     console.log(updatedChat);
74     res.redirect("/chats");
75 });
76
```

```
const Chat = require("../models/chat.js");  
const methodOverride = require("method-override");
```



```
const chat = require( './models/chat.js' );  
const methodOverride = require("method-override");  
  
app.set("views", path.join(__dirname, "views"));  
app.set("view engine", "ejs");  
app.use(express.static(path.join(__dirname, "public")));  
app.use(express.urlencoded({ extended: true }));  
app.use(methodOverride("_method"));
```

//Update Route

```
app.put("/chats/:id", async (req, res) => {  
  ⚡let { id } = req.params;  
  let { msg: newMsg } = req.body;  
  console.log(newMsg);  
  let updatedChat = await Chat.findByIdAndUpdate(  
    id,  
    { msg: newMsg },  
    { runValidators: true, new: true }  
  );  
  console.log(updatedChat);  
  res.redirect("/chats");  
});
```

Edit your chat: 6505f5506e3a3c098d6c907d

This chat was sent from **neha** to **preeti**

send me notes for the practical exam

Edit

From : *neha*

send me notes for the practical exam

edit

Received by *preeti*

00:04:56

Sun-Sep-17-2023

Chats

Destroy Route



DELETE /chats/:id

EXPLORER

...

JS index.js

<> index.ejs

JS chat.js

▼ MONGODB

▼ models

JS chat.js

> node_modules

> public

▼ views

<> edit.ejs

<> index.ejs

<> new.ejs

JS index.js

JS init.js

{ } package-lock.json

{ } package.json

views > <> index.ejs > html > body > ? > div.chat > div.msg > form

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

<% for(let chat of chats) { %>

<div class="chat">

<p>From : <i><%= chat.from %></i></p>

<div class="msg">

<p><%= chat.msg %></p>

<form method="GET" action="/chats/<%= chat._id %>/edit">

<button>edit</button>

</form>

<form method="POST" action="/chats/<%= chat._id %>?method=DELETE">

<button>delete</button>

</form>

</div>

<p>Received by<i> <%= chat.to %></i></p>

<hr />

<p><%= chat.created_at.toString().split(" ")[4] %></p>

<p><%= chat.created_at.toString().split(" ").slice(0, 4).join("-") %></p>

</div>

From : *amit*

all the best!

edit

delete

Received by *sumit*

00:04:56

Sun-Sep-17-2023

//Destroy Route

```
app.delete("/chats/:id", async (req, res) => {  
  let { id } = req.params;  
  let deletedChat = await Chat.findByIdAndDelete(id);  
  console.log(deletedChat);  
  res.redirect("/chats");  
});
```


connection successful

```
{  
  _id: new ObjectId("6505f5506e3a3c098d6c907d"),  
  from: 'neha',  
  to: 'preeti',  
  msg: 'send me notes for the practical exam',  
  created_at: 2023-09-16T18:34:56.491Z,  
  __v: 0  
}
```