

22/04/2023

todo.js → Some additions

```
const {getTodo} = require("../controller/getTodo")  
router.get("getTodos", getTodo); → mapping
```

getTodo.js → Create the controller

1) import the model

```
const Todo = require("../models/Todo");
```

2) Define route handler

```
exports.getTodo = async (req, res) => {
```

```
  try {
```

function in mongoose

```
    const todos = await Todo.find({});
```

```
    res.status(200).json({
```

```
      {
```

```
        success: true,
```

```
        data: todos,
```

```
        message: "Entire Todo fetched",
```

```
      }  
    });
```

```
  }  
}
```

```
  catch (err) {
```

```
    console.log(err);
```

```
    res.status(500).json({
```

Same as
done in
create
Todo.

→ Same as done in create Todo

```

    success : false,
    error : err.message,
    message : "Server error"
  });
}
}

```

→ empty condition will fetch

Note → Todo.find({}); all the data.

* Important command to solve error.

brew services start mongodb-community

Get single Todo

Pass the id as parameter in the URL.

http://localhost:3000/todo/:id

At the end

getTodo.js

```

exports.getTodoById = async (req, res) => {

```

```

  try {

```

```

    const id = req.params.id;

```

```

    const todo = await Todo.findById
    ( { _id : id } )

```

↳ By default - id in MongoDB.

```

    if (!todo) {

```

```

      return res.status(404).json({

```

```

        success : false,

```

```
    message : "No data found",  
  }  
)  
}
```

//Data found

```
res.status(200).json(  
  {  
    success : true,  
    data : todo,  
    message : `Todo ${id} fetched`  
  }  
)
```

```
}
```

catch block + same

todo.js

```
const {getTodoById} = require("../controller/  
getTodo");
```

```
router.get("/:id", getTodoById);
```

update Todo.js

```
try { //2nd way of extracting id
```

```
  const {id} = req.params;
```

```
  const {title, description} = req.body;
```

function

```
  const todo = await Todo.findByIdAndUpdate
```

```
  (
```

```
    { _id : id },
```

```
    { title, description, updatedAt : Date.now() }  
  )
```



```
res.status(200).json({  
  --- } Same  
  ---  
})  
)
```

Catch → same as others done before

todo.js

```
const {updateTodo} = require("../controller/  
updateTodo");  
router.put("/updateTodo/:id", updateTodo);
```

deleteTodo.js

Here inside try block we will be using the function as

```
await Todo.findByIdAndDelete(id);  
Rest the code is same.
```

todo.js

```
const {deleteTodo} = require("../controller/  
deleteTodo");  
router.delete("/deleteTodo/:id", deleteTodo);
```