Index.js …

*const* express = *require*("express");

*const* connector = *require*("./data/connection");

*const* router = *require*("./data/TodoHandler");

*const* app = *express*();

app.*use*(express.*json*());

*connector*();

app.*use*("/todo", router);

app.*listen*(3000, () *=>* {

  console.*log*("Listenin on port 3000");

});

.env

TODO\_URL = "mongodb://localhost:27017/todos"

Package.json …

{

  "scripts": {

    "start": "nodemon index.js",

    "production": "NODE\_ENV=production nodemon index.js"

  },

  "dependencies": {

    "body-parser": "^1.20.0",

    "cookie-parser": "^1.4.6",

    "cors": "^2.8.5",

    "dotenv": "^16.0.3",

    "ejs": "^3.1.8",

    "express": "^4.18.1",

    "mongoose": "^6.7.0",

    "multer": "^1.4.5-lts.1"

  }

}

Now inside ‘data’ folder.

Connection.js …

*require*("dotenv").*config*();

*const* mongoose = *require*("mongoose");

*const* db\_url = process.env.TODO\_URL;

*const* *connector* = () *=>* {

  mongoose

    .*connect*(db\_url)

    .*then*(() *=>* console.*log*("Connection with databse established"))

    .*catch*((*err*) *=>* {

      console.*log*(err);

    });

};

module.exports = connector;

TodoSchema.js …

*const* mongoose = *require*("mongoose");

*const* todoSchema = new mongoose.*Schema*({

  title: {

    type: String,

    required: true,

  },

  description: String,

  status: {

    type: String,

    enum: ["active", "inactive"],

  },

  date: {

    type: Date,

    default: Date.*now*(),

  },

});

module.exports = todoSchema;

TodoHandler.js …

*const* express = *require*("express");

*const* router = express.*Router*();

*const* todoSchema = *require*("./TodoSchema");

*const* mongoose = *require*("mongoose");

*const* Todo = mongoose.*model*("Todo", todoSchema);

*// get multiple todos*

router.*get*("/getmultiple", async (*req*, *res*) *=>* {

*//res.send("Get multiple todos");*

  try {

*//const result1 = await Todo.find({});*

*const* result2 = *await* Todo.*find*({ status: "active" });

    res.*status*(200).*json*(result2);

  } catch (error) {

    res.*status*(500).*json*({ error: error });

  }

});

*//get a todo by id*

router.*get*("/get/:id", async (*req*, *res*) *=>* {

*//res.send("Get a todo");*

  try {

*const* id = req.params.id;

*const* result = *await* Todo.*findById*(id);

    res.*status*(200).*json*(result);

  } catch (error) {

    res.*status*(500).*json*({ error: error });

  }

});

*// post a todo*

router.*post*("/createone", async (*req*, *res*) *=>* {

*//res.send("create one todo");*

  try {

*const* todo = new *Todo*({

      title: req.body.title,

      description: req.body.description,

      status: req.body.status,

      date: req.body.date,

    });

*const* result = *await* todo.*save*();

    res.*status*(200).*json*(result);

  } catch (error) {

    res.*status*(500).*json*({ error: error });

  }

});

*// post multiple todos*

router.*post*("/createmultiple", async (*req*, *res*) *=>* {

*//res.send("crete multiple todos");*

  try {

*const* result = *await* Todo.*insertMany*(req.body);

    res.*status*(200).*json*(result);

  } catch (error) {

    res.*status*(500).*json*({ error: error });

  }

});

*// update a todo by id*

router.*put*("/updatebyid/:id", async (*req*, *res*) *=>* {

  try {

*const* id = req.params.id;

*const* updateData = { $set: { status: "active" } };

*const* options = { new: true };

*const* result = *await* Todo.*findByIdAndUpdate*(id, updateData, options);

    res.*status*(200).*json*(result);

  } catch (error) {

    res.*status*(500).*json*({ error: error });

  }

});

*// update multiple todos*

router.*put*("/updatemultiple", async (*req*, *res*) *=>* {

*//res.send("update multiple todos");*

  try {

*const* filterData = { title: "JAVASCRIPT" };

*const* updateData = { status: "active" };

*const* result = *await* Todo.*updateMany*(filterData, updateData);

    res.*status*(200).*json*(result);

  } catch (error) {

    res.*status*(500).*json*({ error: error });

  }

});

*// delete a todo by id*

router.*delete*("/delete/:id", async (*req*, *res*) *=>* {

  try {

*const* id = req.params.id;

*const* result = *await* Todo.*findByIdAndDelete*(id);

    res.*status*(200).*json*({ message: "deleted successfully" });

  } catch (error) {

    res.*status*(500).*json*({ error: error });

  }

});

*// delete multiple todos*

router.*delete*("/deletemultiple", async (*req*, *res*) *=>* {

*//res.send("delete multiple todos");*

  try {

*const* filterdata = { status: "active" };

*const* result = *await* Todo.*deleteMany*(filterdata);

    res.*status*(200).*json*({ message: "multiple todos deleted" });

  } catch (error) {

    res.*status*(500).*json*({ error: error });

  }

});

module.exports = router;