Wings1 T1 Mini-project - Nodejs APIs

Open **Project** folder (you can find in desktop) in VS Code application

Nodejs:

Type below code in **"/Desktop/Projects/wings.../Nodejs/src/routers/sellandBuy.js"** file:

```javascript
const e = require("express");
const express = require("express");
const SellBuy = require("../mongoose/models/sellBuy")
const router = express.Router()

router.get("/", async (req, res) => {
    try {
        if (req.query.product) {
            const data = await SellBuy.find({ productName: req.query.product })
            res.status(200).json(data)
        } else if (req.query.sortBy) {
            let list = req.query.sortBy
            let ord = -1;
            if (list.charAt(0) == "l") {
                ord = 1
            }
            if (list.includes("S")) {
                const data = await SellBuy.find().sort({ soldPrice: ord })
                res.status(200).json(data)
            } else {
                const data = await SellBuy.find().sort({ costPrice: ord })
                res.status(200).json(data)
            }
        } else {
            const data = await SellBuy.find()
            res.status(200).json(data)
        }
    } catch (err) {
        res.status(200).json(data)
    }
})

router.post("/", async (req, res, next) => {
    const data = new SellBuy(req.body)
    try {
        if (data.productName.length < 4) {
            const err = new Error("product name should have minimum of four
characters")
```

```javascript
                err.status = 400
                next(err)
            } else if (data.costPrice <= 0) {
                const err = new Error("cost price value cannot be zero or negative
value")
                err.status = 400
                next(err)

            } else if (data.soldPrice <= 0) {
                const err = new Error("sold price value cannot be zero or negative
value")
                err.status = 400
                next(err)

            } else {
                await data.save();
                res.status(201).json({ message: "Product Added" })
            }
        } catch (err) {
            res.status(200).json(data)
        }
})


router.patch("/:id", async (req, res, next) => {

        try {
            const data = req.body.soldPrice;
            if (data <= 0) {
                const err = new Error("sold price value cannot be zero or negative
value")
                err.status = 400
                next(err)
            } else {
                res.status(200).json({ message: "Updated Successfully" })
            }
        } catch (err) {
            res.status(200).json(data)
        }
})

router.delete("/:id", async (req, res) => {
        try {
            let id = req.params.id
            if (id === undefined || id === '' || id === null) {
```

```
            res.status(400).json(data)
        }
        const data = await SellBuy.findByIdAndDelete(id)
        res.status(200).json({ message: "Deleted successfully" })
    } catch (err) {
        res.status(400).json({ error: err })
    }
})
```

Type below code in "/Desktop/Projects/wings.../ Nodejs /src/app.js" file:

```
const express = require("express");
const sellAndBuyRouter = require("./routers/sellAndBuy");
const bodyParser = require("body-parser");
require("./mongoose/connect_db/mongoose");

const app = express();
app.use(express.json());
app.use(bodyParser.json());
app.use((req, res, next) => {
    res.header("Access-Control-Allow-Origin", "*");
    res.header("Access-Control-Allow-Headers", "*");
    if (req.method === "OPTIONS") {
        res.header("Access-Control-Allow-Methods", "GET,DELETE,PATCH");
        return res.status(200).json({});
    }
    next();
});
app.use('/sellProduct', sellAndBuyRouter);
app.use((error, req, res, next) => {
    res.status(error.status || 500)
    res.json({ error: error.message })
})

module.exports = app;
```

Install project dependencies by using "sh install.sh" command in the terminal and to run the application by using "sh run.sh"
To test the application "sh test.sh "