App.js
```
const express = require("express")
const mongoose = require("mongoose")

var url = "mongodb://127.0.0.1:27017/RestApiDB"

const app = express()

mongoose.connect(url)

const con = mongoose.connection

con.on("open", ()=>{
console.log("Connected to Database")
})

app.use(express.json())

constemployeeRouter = require("./routes/employees")

app.use("/employees",employeeRouter)

app.listen(8084,()=>{

console.log("Server started")

})
```
Employees.js
```
const express = require("express")
const router = express.Router();

const Employee = require("../models/employee");
//const employee = require("../models/employee");

router.get("/", async(req,res)=>{
try{
const employees = await Employee.find()
res.json(employees)
}
catch(err)
{
res.send("Error is:"+err)
}
})

router.post("/", async(req,res)=>{
const employee = new Employee({
name : req.body.name,
designation :req.body.designation
})

try{
const e1 = await employee.save()
res.send(e1)
res.json(e1)
}
```

```
catch(err)
{
res.send("Error is:"+err)
}
})

router.patch('/update/:id', async (req, res) => {
try {
const id = req.params.id;
constupdatedData = req.body;
const options = { new: true };

const result = await employee.findByIdAndUpdate(
id, updatedData, options
)

res.send(result)
}
catch (error) {
res.status(400).json({ message: error.message })
}
})

router.delete('/delete/:id', async (req, res) => {
try {
const id = req.params.id;
const data = await employee.findByIdAndDelete(id)
res.send(`Document with ${data.name} has been deleted..`)
}
catch (error) {
res.status(400).json({ message: error.message })

}
})

module.exports = router
```

employee.js
```
const mongoose = require("mongoose")

constemployeeSchema = new mongoose.Schema({
name : String,
designation : String
})

module.exports = mongoose.model("Employee", employeeSchema)
```