Extra Assignment

Name: Rana Aayurshi VijayKumar

RollNo.: 39

Subject: Application Development using FullStack(701)

Semester: 7th

Question:

Design and develop a module to manage vehicle data for a vehicle resale agency considering following requirements.

- Design two collections

o (1) vehicle_category (c_id, category_name)

o (2) vehicle_data (_id, vehicle_brand, category_name, vehicle picture, price, depreciation , number of years, total price)

- Insert vehicle

o upload a picture

o Load category data in a combo

o Establish relation between two collections

- List vehicle data along with category_name.

Solution:

→app.js

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

```
const vehRoute = require("./Route/VehRoute");
const dbCon = require("./DbConnection/dbCon");
app.use(express.urlencoded({extended:false}));
app.use(express.json());
app.use(express.static("./uploads"));
app.use("/",vehRoute);
app.set("view engine","hbs");
const start = ()=>{
    try{
            dbCon("mongodb://localhost:27017/vehicleDb");
            var server = app.listen(8000,()=>{
                console.log("Server is listening on port " +
server.address().port);
            });
    catch(err)
        console.log("Error : " + err);
};
start();
```

→mongoDb Connection:

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

const Connection = (url)=>{
    try{
        mongoose.connect(url);
        console.log("Mongodb connected successfully");
    }
    catch(err)
    {
        console.log("Can not connect mongoDb\n" + err);
    }
};

module.exports = Connection;
```

→Models:

- category.js

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

const categorySchema = new mongoose.Schema({
    category_name : {
        type:String
    }
});

const categoryModel =
mongoose.model("Categories", categorySchema);
module.exports = categoryModel;
```

-vehicleData.js

```
const mongoose = require("mongoose");
const vehSchema = new mongoose.Schema({
    vehicle brand:{
        type : String
    category_name : {
       type:String
    },
    vehicle_picture : {
        type:String
    },
    price : {
        type:Number
    depreciation:{
        type : Number
    },
    noOfYears : {
        type:Number
    },
    totalPrice : {
       type:Number
```

```
},
});

const vehModel = mongoose.model("Vehicles", vehSchema);

module.exports = vehModel;
```

→Route:

vehRoute.js

```
const vehicles = require("../Models/vehicleData");
const categories = require("../Models/category");
const vehRoute = require("express").Router();
const multer = require("multer");
var options = multer.diskStorage({
    destination : function(req,file,cb){
        cb(null,'./uploads');
    },
    filename : function(req,file,cb){
        cb(null,file.originalname );
});
var upload = multer({storage:options});
vehRoute.get("/",async(req,res)=>{
    try{
        var data = await vehicles.find({});
        console.log("Vehicles :\n" + data);
        res.render("home", {data:data});
    catch(err)
        console.log(err);
});
```

```
vehRoute.post("/add",upload.single("vehicle picture"),async(req,r
es)=>{
    try{
            let vehPrice = req.body.price;
            for(let i=0;i<req.body.noOfYears;i++)</pre>
            {
                var temp = vehPrice * req.body.depreciation/100;
                vehPrice = vehPrice - temp;
            }
            var data = new vehicles({
                    vehicle brand :req.body.vehicle brand,
                    category name : req.body.category name,
                    vehicle picture : req.file.filename,
                    price : req.body.price,
                    depreciation : req.body.depreciation,
                    noOfYears : req.body.noOfYears,
                    totalPrice : vehPrice
            });
            console.log(req.file)
            console.log("New data : \n" + data);
            var saveData = vehicles.create(data);
            console.log("Vehicle added successfully");
            res.redirect("/");
    catch(err)
        console.log("Error\n" + err);
});
vehRoute.get("/add",async(req,res)=>{
    try{
        var data = await categories.find({});
        res.render("AddNew", {data:data});
    catch(err)
        console.log(err);
```

```
module.exports = vehRoute;
```

→views:

-AddNew.hbs:

```
<html>
    <head>
        <title>Add new vehicle</title>
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet">
    </head>
    <body>
        <form method="post" enctype="multipart/form-data">
            <div class="form-group col-md-6">
              <h6><label>Vehicle Brand</label></h6>
              <input type="text" class="form-control" name="vehicle_brand"</pre>
placeholder="Vehicle Brand">
            </div>
            <div class="form-group col-md-4">
              <h6><label>Vehicle category</label></h6>
              <select name="category_name" class="form-control">
                {{#each data}}
                    <option>{{this.category_name}}</option>
                {{/each}}
              </select>
            </div>
            <div class="form-group col-md-6">
              <h6><label>Vehicle Picture</label></h6>
              <input type="file" class="form-control" name="vehicle_picture">
            </div>
            <div class="form-group col-md-6">
              <h6><label>Price</label></h6>
              <input type="number" class="form-control" name="price"</pre>
placeholder="Price">
            </div>
            <div class="form-group col-md-6">
              <h6><label>Depreciation</label></h6>
              <input type="number" class="form-control" name="depreciation"</pre>
placeholder="Depreciation">
```

-home.hbs:

```
<html>
      <title>Display Vehicles</title>
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet">
   </head>
   <body>
      <h3><a href="/add" class="link-primary" style="text-decoration:</pre>
none;">Add New</a></h3>
      <thead>
         Vehicle brand
            Category name
            Picture
            Price
            Depreciation %
            No of years
            Total price
         </thead>
         {{#each data}}
         {{this.vehicle_brand}}
            {{this.category_name}}
            <img src="{{this.vehicle_picture}}" width="200">
            {{this.price}}
            {{this.depreciation}}
            {{this.noOfYears}}
```

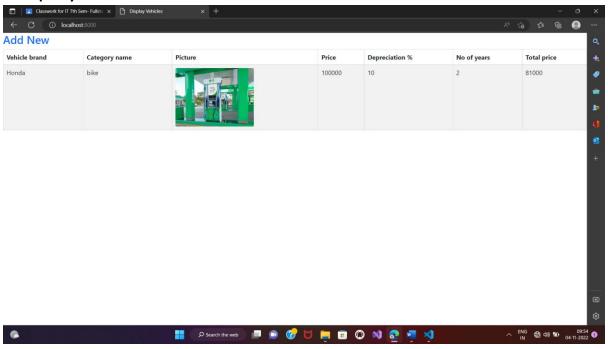
```
{this.totalPrice}}

{{/each}}

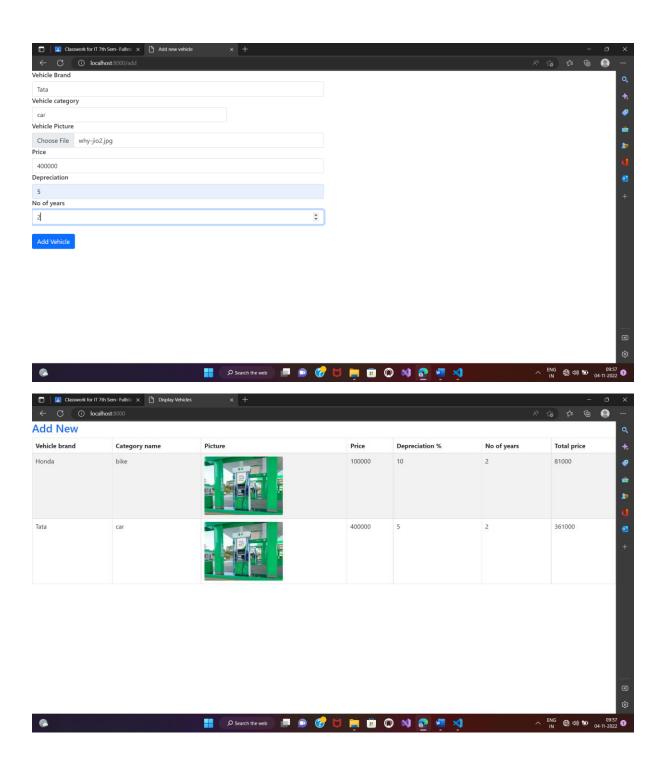
</body>
</html>
```

Output:

Display:



AddNew:



//39 Aayurshi Rana//