/controllers/campgrounds.js

const Campground = require('../models/campground');

const { cloudinary } = require('../cloudinary');

module.exports.index = async (req, res) => {

    const campgrounds = *await* Campground.find();

    res.render('campgrounds/index', { campgrounds })

};

module.exports.renderNewForm = (req, res) => {

    res.render('campgrounds/new');

}

module.exports.renderEditForm = async (req, res) => {

    const { id } = req.params;

    const campground = *await* (Campground.findById(id))

    res.render('campgrounds/edit', { campground });

}

module.exports.updateCampground = async (req, res) => {

    const { id } = req.params;

    const img = req.files.map(f => ({ url: f.path, filename: f.filename }));

*// console.log(id);*

    const cmpgrd = *await* Campground.findByIdAndUpdate(id, req.body.campground, { runValidators: true })

    cmpgrd.images.push(...img)

*await* cmpgrd.save();

*if* (req.body.deleteImg) {

*for* (let imge of req.body.deleteImg) {

*await* cloudinary.uploader.destroy(imge);

        }

*await* camgrd.updateAndOne({ $pull: { images: { filename: { $in: req.body.deleteImg } } } })

    }

    req.flash('success', 'Successfully update Campground');

    res.redirect(`/campgrounds/${cmpgrd.\_id}`);

}

module.exports.showCampground = async (req, res, next) => {

    const { id } = req.params;

    const campground = *await* Campground.findById(id).populate({

        path: 'reviews',

        populate: {

            path: 'author'

        }

    }).populate('author');

*// console.log(campground);*

*if* (!campground) {

        req.flash('error', 'Campground not found');

        res.redirect('/campgrounds');

    }

    res.render('campgrounds/show', { campground });

}

module.exports.createCampground = async (req, res, next) => {

    const { campground } = req.body;

    const campgrounds = new Campground(campground);

    campgrounds.images = req.files.map(f => ({ url: f.path, filename: f.filename }));

    campgrounds.author = req.user.\_id;

*await* campgrounds.save();

    console.log(campgrounds);

    req.flash('success', 'Successfully created new Campground');

    res.redirect(`/campgrounds/${campgrounds.\_id}`);

}

module.exports.deleteCampground = async (req, res) => {

    const { id } = req.params;

*await* Campground.findByIdAndDelete(id);

    req.flash('success', 'Successfully delete Campground');

    res.redirect('/campgrounds');

}

views/campgrounds/show.ejs

<link *rel*="stylesheet" *href*="../stylesheet/stars.css">

<% layout('layouts/boilerplate') %>

    <div *class*="card">

        <div *class*="row" *mb-3*>

            <div *class*="col-md-6">

                <div *id*="carouselCampground" *class*="carousel slide" *data-bs-ride*="carousel">

                    <div *class*="carousel-inner">

                        <% campground.images.forEach((img, i)=> {%>

                            <div *class*="carousel-item <%= i===0? 'active':'' %>">

                                <img *src*="<%=img.url %>" *class*="d-block w-100" *alt*="...">

                            </div>

                            <% }) %>

                    </div>

                    <% *if* (campground.images.length> 1) {%>

                        <button *class*="carousel-control-prev" *type*="button" *data-bs-target*="#carouselCampground"

*data-bs-slide*="prev">

                            <span *class*="carousel-control-prev-icon" *aria-hidden*="true"></span>

                            <span *class*="visually-hidden">Previous</span>

                        </button>

                        <button *class*="carousel-control-next" *type*="button" *data-bs-target*="#carouselCampground"

*data-bs-slide*="next">

                            <span *class*="carousel-control-next-icon" *aria-hidden*="true"></span>

                            <span *class*="visually-hidden">Next</span>

                        </button>

                        <% } %>

                </div>

*<% /\* %>*

*<% for(let img of campground.images) {%>*

*<img src="<%= img.url %>" class="card-img-top" alt="...">*

*<% } %>*

*<% \*/ %>*

                                <div *class*="card-body">

                                    <div *class*="card-title">

                                        <h4>

                                            <%= campground.title %>

                                        </h4>

                                    </div>

                                    <ul *class*="list-group list-group-flush">

                                        <li *class*="list-group-item">

                                            <%= campground.location %>

                                        </li>

                                        <li *class*="list-group-item"> Submitted by <%= campground.author.username %>

                                        </li>

                                        <li *class*="list-group-item">$<%= campground.price %> per day</li>

                                        <li *class*="list-group-item">

                                            <%= campground.description %>

                                        </li>

                                    </ul>

                                    <% *if* (currentUser && campground.author.equals(currentUser.\_id)) { %>

                                        <form *action*="/campgrounds/<%= campground.\_id %>?\_method=DELETE" *method*="POST">

                                            <button *class*="btn btn-danger">Delete</button>

                                        </form>

                                        <% } %>

                                </div>

            </div>

            <div *class*="col-md-6">

                <% *if* (currentUser) {%>

                    <div *class*="text-center">

                        <h2>Leaves a Review</h2>

                    </div>

                    <form *action*="/campgrounds/<%= campground.\_id %>/reviews" *method*="POST" *class*="validate-form"

*novalidate*>

                        <div *class*="card-text mb-3">

*<!-- <div class="mb-3"> -->*

*<!-- <label for="rating" class="form-label">*

*<h5>Enter a Rating</h5>*

*</label> -->*

*<!-- <input type="range" class="form-range" min="1" max="5" id="rating" name="review[rating]"> -->*

                            <fieldset *class*="starability-basic" *id*="rating">

                                <legend>Enter rating:</legend>

                                <input *type*="radio" *id*="no-rate" *class*="input-no-rate" *name*="review[rating]" *value*="1"

*checked* *aria-label*="No rating." />

                                <input *type*="radio" *id*="first-rate1" *name*="review[rating]" *value*="1" />

                                <label *for*="first-rate1" *title*="Terrible">1 star</label>

                                <input *type*="radio" *id*="first-rate2" *name*="review[rating]" *value*="2" />

                                <label *for*="first-rate2" *title*="Not good">2 stars</label>

                                <input *type*="radio" *id*="first-rate3" *name*="review[rating]" *value*="3" />

                                <label *for*="first-rate3" *title*="Average">3 stars</label>

                                <input *type*="radio" *id*="first-rate4" *name*="review[rating]" *value*="4" />

                                <label *for*="first-rate4" *title*="Very good">4 stars</label>

                                <input *type*="radio" *id*="first-rate5" *name*="review[rating]" *value*="5" />

                                <label *for*="first-rate5" *title*="Amazing">5 stars</label>

                            </fieldset>

                        </div>

                        <div *class*="card-text mb-3">

                            <label *for*="body" *class*="form-label">

                                <h5>Write a review</h5>

                            </label>

                            <textarea *name*="review[body]" *id*="body" *cols*="30" *rows*="5" *class*="form-control "

*required*></textarea>

                        </div>

                        <div *class*="card-text mb-3">

                            <button *class*="btn btn-primary">Submit</button>

                        </div>

                    </form>

                    <% } %>

                        <% *for*(let review of campground.reviews) {%>

                            <div *class*="card-body">

*<!-- <h5 class="card-title mb-3">*

*Rating : <%= review.rating %>*

*</h5> -->*

                                <p *class*="starability-result" *data-rating*="<%= review.rating %>">

                                    Rated: <%= review.rating %> stars

                                </p>

                                <div *class*="card-subtitle mb-3">

                                    Contributted by : <%= review.author.username %>

                                </div>

                                <div *class*="card-text mb-3">

                                    Review : <%= review.body %>

                                </div>

                                <% *if* (currentUser && review.author.equals(currentUser.\_id)) {%>

                                    <form

*action*="/campgrounds/<%= campground.\_id %>/reviews/<%= review.\_id %>?\_method=DELETE"

*method*="POST" *mb-3*>

                                        <button *class*="btn btn-danger btn-sm">Delete</button>

                                    </form>

                                    <% } %>

                            </div>

                            <% } %>

            </div>

        </div>

    </div>

    </div>

*<!-- <h1>*

*<%= campground.title %>*

*</h1>*

*<h2>*

*<%= campground.location %>*

*</h2>*

*<p>*

*<img src="<%= campground.image %>" alt="">*

*</p>*

*<p>*

*$<%= campground.price %> per day*

*</p>*

*<p>*

*<%= campground.description %>*

*</p>*

*<form action="/campgrounds/<%= campground.\_id %>?\_method=DELETE" method="POST">*

*<button>Delete</button>*

*</form>*

*<footer>*

*<p>*

*<a href="/campgrounds">Back to Full List</a>*

*</p>*

*<p>*

*<a href="/campgrounds/<%= campground.\_id %>/edit">Edit Campground</a>*

*</p>*

*</footer> -->*

/views/campgrounds/edit.ejs

<% layout('layouts/boilerplate') %>

    <div *class*="row">

        <div *class*="col-md-6 offset-3">

            <h1 *class*="text-center">

                Edit Campground

            </h1>

            <form *action*="/campgrounds/<%= campground.\_id %>?\_method=PUT" *method*="POST" *novalidate* *class*="validate-form"

*enctype*="multipart/form-data">

                <div *class*="mb-3">

                    <label *for*="title" *class*="form-label">

                        Campground title

                    </label>

                    <input *type*="text" *class*="form-control" *id*="title" *name*="campground[title]"

*value*="<%=campground.title %>" *required*>

                </div>

                <div *class*="mb-3">

                    <label *for*="location" *class*="form-label">

                        Location

                    </label>

                    <input *type*="text" *class*="form-control" *id*="location" *name*="campground[location]"

*value*="<%=campground.location %>" *required*>

                </div>

                <div *class*="mb-3">

                    <label *for*="price" *class*="form-label">

                        Campground Price

                    </label>

                    <div *class*="input-group">

                        <span *class*="input-group-text">$</span>

                        <input *type*="number" *class*="form-control" *name*="campground[price]" *aria-label*="Price per Night"

*id*="price" *value*="<%=campground.price %>" *required*>

                        <span *class*="input-group-text">.00</span>

                    </div>

                </div>

                <div *class*="mb-3">

                    <label *for*="description" *class*="form-label"> Campground Description</label>

                    <textarea *name*="campground[description]" *id*="description" *cols*="30" *rows*="5" *class*="form-control"

*required*>

                    <%=campground.description %>

                </textarea>

                </div>

                <div *class*="mb-3">

                    <label *for*="image" *class*="form-label">Add Image(s)</label>

                    <input *class*="form-control" *type*="file" *name*="image" *id*="image" *multiple*>

                </div>

                <div *class*="mb-3">

                    <% campground.images.forEach(function(img, i) { %>

                        <img *src*="<%=img.url %>" *alt*="" *class*="img-thumbnail">

                        <div *class*="form-check-inline">

                            <input *type*="checkbox" *id*="image-<%= i %>" *name*="deleteImg[]" *value*="<%= img.filename %>">

                        </div>

                        <label *for*="image-<%= i %>"> Delete?</label>

                        <% }) %>

                </div>

                <% *if* (currentUser && campground.author.equals(currentUser.\_id)) {%>

                    <div *class*="mb-3">

                        <button *class*="btn btn-success">Submit</button>

                    </div>

                    <% } %>

            </form>

        </div>

    </div>

    //

*<!-- <form action="/campgrounds/<%= campground.\_id %>?\_method=PUT" method="POST">*

*<div>*

*<label for="title">Campground title</label>*

*<input type="text" id="title" name="campground[title]" value="<%=campground.title %>">*

*</div>*

*<div>*

*<label for="image">Image</label>*

*<input type="text" name="campground[image]" id="image" value="<%=campground.image %>">*

*</div>*

*<div>*

*<label for="location">Location</label>*

*<input type="text" name="campground[location]" id="location" value="<%=campground.location %>">*

*</div>*

*<div>*

*<label for="price">Price</label>*

*<input type="number" name="campground[price]" id="price" value="<%=campground.price %>">*

*</div>*

*<div>*

*<label for="description">Description</label>*

*<textarea name="campground[description]" id="description" cols="30" rows="10">*

*<%=campground.description %>*

*</textarea>>*

*</div>*

*<button>Submit</button>*

*</form>*

*<footer>*

*<p>*

*<a href="/campgrounds">Back to All Campgrounds</a>*

*</p>*

*</footer> -->*

views/campgrounds/index.ejs

<% layout('layouts/boilerplate') %>

*<!-- <h1>All Campgrounds</h1> -->*

    <% *for* (let campground of campgrounds) { %>

        <div *class*="card">

            <div *class*="row" *mb-3*>

                <div *class*="col-md-4">

                    <img *src*="<%= campground.images[0].url %>" *alt*="" *class*="img-fluid">

                </div>

                <div *class*="col-md-8">

                    <div *class*="card-body">

                        <div *class*="card-title">

                            <h5>

                                <%= campground.title %>

                            </h5>

                        </div>

                        <div *class*="card-text">

                            <div>

                                <%= campground.description %>

                            </div>

                        </div>

                        <div *class*="card-text">

                            <div>

                                $<%= campground.price %> per night

                            </div>

                        </div>

                        <small *class*="text-muted">

                            <%= campground.location %>

                        </small>

                        <div *class*="in-line">

                            <a *href*="/campgrounds/<%= campground.\_id %>" *class*="btn btn-primary">

                                Show Campground

                            </a>

                            <% *if* (currentUser && campground.author.equals(currentUser.\_id)) { %>

                                <a *href*="/campgrounds/<%= campground.\_id %>/edit" *class*="btn btn-info">

                                    Edit Campground

                                </a>

                                <% } %>

                        </div *class*="in-line">

                    </div>

                </div>

                <% } %>

            </div>

        </div>

*<!-- <ul>*

*<% for (let campground of campgrounds) { %>*

*<li>*

*<a href="/campgrounds/<%= campground.\_id %>">*

*<%= campground.title %>*

*</a>*

*<a href="/campgrounds/<%= campground.\_id %>/edit">Edit Campground</a>*

*</li>*

*<% } %>*

*</ul>*

*<footer>*

*<a href="/campgrounds/new">New Campgrounds</a>*

*</footer> -->*

/routes/campgrounds.ejs

const express = require('express');

const router = express.Router();

const catchAsync = require('../utils/catchAsync');

const campgrounds = require('../controllers/campgrounds')

const { reqLogin, isAuthor, validateCampground } = require('../middleware');

const multer = require('multer');

const { storage } = require('../cloudinary');

const upload = multer({ storage });

router.route('/')

    .get(catchAsync(campgrounds.index))

    .post(reqLogin, upload.array('image'), validateCampground, catchAsync(campgrounds.createCampground))

*// .post(upload.array('image'), (req, res) => {*

*//     console.log(req.body, req.files);*

*//     res.send('image(s) uploaded')*

*// })*

router.get('/new', reqLogin, campgrounds.renderNewForm);

router.route('/:id')

    .put(reqLogin, isAuthor, upload.array('image'), validateCampground, catchAsync(campgrounds.updateCampground))

    .get(catchAsync(campgrounds.showCampground))

    .delete(reqLogin, isAuthor, catchAsync(campgrounds.deleteCampground))

*// router.get('/', catchAsync(campgrounds.index));*

*// router.get('/new', reqLogin, campgrounds.renderNewForm);*

router.get('/:id/edit', reqLogin, isAuthor, catchAsync(campgrounds.renderEditForm));

*// router.put('/:id', validateCampground, reqLogin, isAuthor, catchAsync(campgrounds.updateCampground));*

*// router.get('/:id', catchAsync(campgrounds.showCampground));*

*// router.post('/', validateCampground, reqLogin, catchAsync(campgrounds.createCampground));*

*// router.delete('/:id', reqLogin, isAuthor, catchAsync(campgrounds.deleteCampground));*

module.exports = router;