

```
1 using Blog.Data.FileManager;
2 using Blog.Data.Repository;
3 using Blog.Models;
4 using Blog.ViewModels;
5 using Microsoft.AspNetCore.Authorization;
6 using Microsoft.AspNetCore.Mvc;
7 using System;
8 using System.Collections.Generic;
9 using System.Linq;
10 using System.Text;
11 using System.Threading.Tasks;
12
13 namespace Blog.Controllers
14 {
15     [Authorize(Roles = "Admin")]
16     public class PanelController : Controller
17     {
18         private IRepository _repo;
19         private IFileManager _fileManager;
20
21         public PanelController(
22             IRepository repo,
23             IFileManager fileManager
24         )
25         {
26             _repo = repo;
27             _fileManager = fileManager;
28         }
29
30         public IActionResult Index()
31         {
32             var posts = _repo.GetAllPosts();
33             return View(posts);
34         }
35
36         [HttpGet]
37         public IActionResult Edit(int? id)
38         {
39             if (id == null)
40             {
41                 return View(new PostViewModel());
42             }
43             else
44             {
45                 var post = _repo.GetPost((int)id);
46                 return View(new PostViewModel
47                 {
48                     Id = post.Id,
49                     Title = post.Title,
50                     Body = post.Body
51                 });
52             }
53         }
54
55         [HttpPost]
56         public async Task<IActionResult> Edit(PostViewModel vm)
```

```
57     {
58         var post = new Post
59         {
60             Id = vm.Id,
61             Title = vm.Title,
62             Body = vm.Body,
63             Image = await _fileManager.SaveImage(vm.Image)
64         };
65
66         if (post.Id > 0)
67             _repo.UpdatePost(post);
68         else
69             _repo.AddPost(post);
70
71         if (await _repo.SaveChangesAsync())
72             return RedirectToAction("Index");
73         else
74             return View(post);
75     }
76
77     [HttpGet]
78     public async Task<IActionResult> Remove(int id)
79     {
80         _repo.RemovePost(id);
81         await _repo.SaveChangesAsync();
82         return RedirectToAction("Index");
83     }
84 }
85 }
86
```