

```
1 using Microsoft.AspNetCore.Authorization;
2 using Microsoft.AspNetCore.Hosting;
3 using Microsoft.AspNetCore.Mvc;
4 using Microsoft.Extensions.Logging;
5 using NEAproject.Models;
6 using Newtonsoft.Json;
7 using System;
8 using System.Collections.Generic;
9 using System.Diagnostics;
10 using System.IO;
11 using System.Linq;
12 using System.Threading.Tasks;
13
14
15
16 namespace NEAproject.Controllers
17 {
18     public class HomeController : Controller
19     {
20
21         private readonly ILogger<HomeController> _logger;
22
23
24
25         public HomeController(ILogger<HomeController> logger)
26         {
27             _logger = logger;
28
29
30         }
31
32
33         public ActionResult Index(homemodel homemodel, bool? loginsuccess =  ➤
34             false, string Save = null)
35             //ActionResult adds another task to render the page on the browser ➤
36             //in this instance. bridge between CSS/HTML page and browser
37         {
38             bool valid = true;
39             //to the code
40             if (homemodel.selectcomplexity == "getLineardatapoints")
41             {
42                 ViewBag.datapointlist = JsonConvert.SerializeObject ➤
43                     (Helper.getLineardatapoints(homemodel.valueofn));
44                 valid = Helper.isvalidint(homemodel.valueofn.ToString());
45             }
46             else if (homemodel.selectcomplexity == "getNtothe2points")
47             {
48                 ViewBag.datapointlist = JsonConvert.SerializeObject ➤
49                     (Helper.getNtothe2points(homemodel.valueofn));
50                 valid = Helper.isvalidint(homemodel.valueofn.ToString());
51             }
52             else if (homemodel.selectcomplexity == "get2totheNpoints")
53             {
54                 ViewBag.datapointlist = JsonConvert.SerializeObject ➤
55                     (Helper.get2totheNpoints(homemodel.valueofn));
56                 valid = Helper.isvalidint(homemodel.valueofn.ToString());
57             }
58         }
59     }
60 }
```

```
50         ViewBag.datapointlist = JsonConvert.SerializeObject
           (Helper.get2totheNpoints(homemodel.valueofn));
51         valid = Helper.isvalidint(homemodel.valueofn.ToString());
52     }
53     else
54     {
55         List<datapoint> datapointlist = new List<datapoint>();
56         datapointlist.Add(new datapoint("0", 0));
57         datapointlist.Add(new datapoint("1", 1));
58         datapointlist.Add(new datapoint("2", 2));
59         datapointlist.Add(new datapoint("3", 3));
60         datapointlist.Add(new datapoint("4", 4));
61         datapointlist.Add(new datapoint("5", 5));
62         datapointlist.Add(new datapoint("6", 6));
63         datapointlist.Add(new datapoint("7", 7));
64         datapointlist.Add(new datapoint("8", 8));
65         datapointlist.Add(new datapoint("9", 9));
66         datapointlist.Add(new datapoint("10", 10));
67         datapointlist.Add(new datapoint("11", 11));
68         datapointlist.Add(new datapoint("12", 12));
69         datapointlist.Add(new datapoint("13", 13));
70         datapointlist.Add(new datapoint("14", 14));
71         datapointlist.Add(new datapoint("15", 15));
72
73
74
75         ViewBag.datapointlist = JsonConvert.SerializeObject
           (datapointlist);
76     }
77     ViewBag.validint = valid;
78
79
80
81     return View(homemodel);
82 }
83
84
85
86 public ActionResult Comparison(homemodel homemodel, string Save =
   null)
87 {
88     //to the code
89     ViewBag.Message = "Your comparison page.";
90     if (homemodel != null)
91     {
92
93         ViewBag.datapointlist = JsonConvert.SerializeObject
           (Helper.getLineardatapoints(homemodel.valueofn));
94
95         ViewBag.datapointlist1 = JsonConvert.SerializeObject
           (Helper.getNtothe2points(homemodel.valueofn));
96
97         ViewBag.datapointlist2 = JsonConvert.SerializeObject
```

```
(Helper.get2totheNpoints(homemodel.valueofn));  
98  
99     }  
100  
101  
102     return View(homemodel);  
103 }  
104  
105  
106 public IActionResult About()  
107 {  
108     ViewBag.Message = "Your application description page.";  
109     return View();  
110 }  
111  
112 [ResponseCache(Duration = 0, Location = ResponseCacheLocation.None,  ➤  
    NoStore = true)]  
113 public IActionResult Error()  
114 {  
115     return View(new ErrorViewModel { RequestId =  ➤  
        Activity.Current?.Id ?? HttpContext.TraceIdentifier });  
116 }  
117  
118 }  
119 }  
120
```