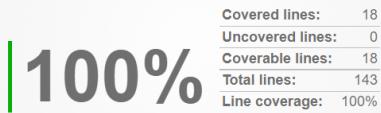


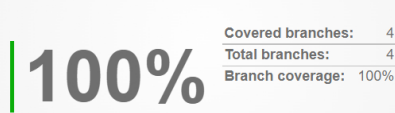
### Information

Class: FuelQuoteApp\_p1.Controllers.QuoteController  
Assembly: FuelQuoteApp\_p1  
File(s): E:\UHSummer\SD\FuelQuoteApp\_p1\FuelQuoteApp\_p1\Controllers\QuoteController.cs

### Line coverage



### Branch coverage



### Method coverage

Method coverage is only available for sponsors.

Upgrade to PRO version

- .ctor(FuelQuoteApp\_p1.Provider.IFuelQu...
- QuoteDataValidation(FuelQuoteApp\_p1....

### Metrics

Method	Branch coverage ⓘ	Cyclomatic complexity ⓘ	Line coverage ⓘ
.ctor(...)	100%	1	100%
QuoteDataValidation(...)	100%	4	100%

### File(s)

E:\UHSummer\SD\FuelQuoteApp\_p1\FuelQuoteApp\_p1\Controllers\QuoteController.cs

```
# Line Line coverage
1 using FuelQuoteApp_p1.BusinessLayer.BL;
2 using FuelQuoteApp_p1.EntModels.Models;
3 using FuelQuoteApp_p1.Models.Quote;
4 using FuelQuoteApp_p1.Provider;
5 using FuelQuoteApp_p1.Repository;
6 using Microsoft.AspNetCore.Http;
7 using Microsoft.AspNetCore.Mvc;
8 using Newtonsoft.Json;
9 using System;
10 using System.Collections.Generic;
11 using System.Diagnostics.CodeAnalysis;
12 using System.Globalization;
13
14 namespace FuelQuoteApp_p1.Controllers
15 {
16     public class QuoteController : Controller
17     {
18         private readonly IFuelQuoteProvider _FuelQuotePro;
19         public QuoteController(IFuelQuoteProvider FuelQuotePro)
20         {
21             _FuelQuotePro = FuelQuotePro;
22         }
23
24         [HttpGet]
25         [ExcludeFromCodeCoverage]
26         public IActionResult GetQuote()
27         {
28             FQuote quote = new FQuote();
29             Client client = new Client();
30             client = JsonConvert.DeserializeObject<Client>(HttpContext.Session.GetString("ClientDetails"));
31             quote.DeliveryAddress = client.Address1 + " " + client.Address2;
32             return View(quote);
33         }
34
35         [HttpPost]
36         [ExcludeFromCodeCoverage]
37         public IActionResult GetQuote(FQuote quote)
38         {
39
40             Price pricetdetails = new Price();
41             pricetdetails = GetPrice(quote);
42             quote.PricePerGallon = pricetdetails.PricePerGallon;
43             quote.TotalAmount = pricetdetails.TotalAmount;
44             HttpContext.Session.SetString("QuoteDetails", JsonConvert.SerializeObject(quote));
45
46             return RedirectToAction("GetFinalQuote");
47         }
48     }
49
50     [HttpGet]
51     [ExcludeFromCodeCoverage]
52     public IActionResult QuoteHistory()
53     {
54
55         int usrID = _FuelQuotePro.GetUserID(User.Identity.Name);
56         IEnumerable<Quote> quotess = _FuelQuotePro.GetQuoteHistory(usrID);
57
58     }
```

```

58         return View(quotess);
59     }
60
61     [ExcludeFromCodeCoverage]
62     public Price GetPrice(FQuote quote)
63     {
64         int usrID = _FuelQuotePro.GetUserID(User.Identity.Name);
65
66         Client client = new Client();
67         client = JsonConvert.DeserializeObject<Client>(HttpContext.Session.GetString("ClientDetails"));
68         QuoteDetails quoteInfo = new QuoteDetails
69         {
70             DateRequested = quote.DateRequested,
71             GallonsRequested = quote.GallonsRequested,
72             State = client.State,
73             quoteHistory = _FuelQuotePro.GetQuoteHistoryCount(usrID)
74         };
75
76         PriceModule getPrice = new PriceModule();
77         Price price = getPrice.GetPrice(quoteInfo);
78
79         return price;
80     }
81
82
83     [HttpGet]
84     [ExcludeFromCodeCoverage]
85     public IActionResult GetFinalQuote()
86     {
87         FQuote quote = new FQuote();
88         Client client = new Client();
89         quote = JsonConvert.DeserializeObject<FQuote>(HttpContext.Session.GetString("QuoteDetails"));
90         client = JsonConvert.DeserializeObject<Client>(HttpContext.Session.GetString("ClientDetails"));
91         quote.DeliveryAddress = client.Address1 + " " + client.Address2;
92         return View(quote);
93     }
94
95
96     [HttpPost]
97     [ExcludeFromCodeCoverage]
98     public IActionResult GetFinalQuote(FQuote quote)
99     {
100         quote = JsonConvert.DeserializeObject<FQuote>(HttpContext.Session.GetString("QuoteDetails"));
101         Client client = new Client();
102         client = JsonConvert.DeserializeObject<Client>(HttpContext.Session.GetString("ClientDetails"));
103         quote.DeliveryAddress = client.Address1 + " " + client.Address2;
104
105         Quote dbquote = new Quote
106         {
107             DateRequested = quote.DateRequested,
108             DeliveryAddress = quote.DeliveryAddress,
109             GallonsRequested = quote.GallonsRequested,
110             PricePerGallon = quote.PricePerGallon,
111             TotalAmount = quote.TotalAmount,
112
113         };
114         User userinfo = new User();
115         dbquote.User_Id = userinfo.Id;
116         dbquote.User_Id = _FuelQuotePro.GetUserID(User.Identity.Name);
117         dbquote = _FuelQuotePro.AddQuote(dbquote);
118         return View("SavedQuote", dbquote);
119     }
120
121
122
123
124     public bool QuoteDataValidation(Quote data)
125     {
126         bool flag = false;
127         if (data.GallonsRequested != 0) /*Default value is 0*/
128         {
129             if (DateTime.Compare(data.DateRequested, DateTime.Now) > 0)
130             {
131                 flag = true;
132             }
133         }
134         else
135         {
136             flag = false;
137         }
138
139         return flag;
140     }
141 }
142
143 }

```