Information

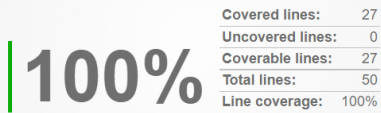**Class:** FuelQuoteApp_p1.BusinessLayer.BL.PriceModule
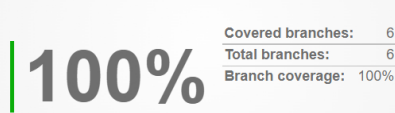**Assembly:** FuelQuoteApp_p1.BusinessLayer
File(s): E:\UH\Summer\SD\FuelQuoteApp_p1\FuelQuoteApp_p1.BusinessLayer\BL\PriceModule.cs

Line coverage

100%

Covered lines:	27
Uncovered lines:	0
Coverable lines:	27
Total lines:	50
Line coverage:	100%

Branch coverage

100%

Covered branches:	6
Total branches:	6
Branch coverage:	100%

Method coverage

Method coverage is only available for sponsors.

Upgrade to PRO version

Metrics

Method	Branch coverage ℹ	Cyclomatic complexity ℹ	Line coverage ℹ
GetPrice(...)	100%	6	100%

File(s)

E:\UH\Summer\SD\FuelQuoteApp_p1\FuelQuoteApp_p1.BusinessLayer\BL\PriceModule.cs

#	Line	Line coverage
	1	namespace FuelQuoteApp_p1.BusinessLayer.BL
	2	{
	3	public class PriceModule
	4	{
	5	private const float PricePerGallon = 1.5f;
	6	private const float CompanyProfitFactor = 0.1f;
	7	
	8	public Price GetPrice(QuoteDetails quoteinfo)
4	9	{
4	10	float locationFactor = 0.0f;
4	11	float rateHistoryFactor = 0.0f;
4	12	float gallonsRequestedFactor = 0.0f;
	13	
4	14	Price finalPrices = new Price();
	15	
	16	//For TEXAS State the location factor is 2% else for any other state location factor is 4%
4	17	if (quoteinfo.State == "TX")
2	18	{
2	19	locationFactor = 0.02f;
2	20	}
	21	else
2	22	{
2	23	locationFactor = 0.04f;
2	24	}
	25	
	26	//Rate History Factor = 1% if client requested fuel before, 0% if no history
4	27	if (quoteinfo.quoteHistory > 0)
2	28	{
2	29	rateHistoryFactor = 0.01f;
2	30	}
	31	
	32	//Gallons Requested Factor = 2% if more than 1000 Gallons, 3% if less
4	33	if (quoteinfo.GallonsRequested > 1000)
2	34	{
2	35	gallonsRequestedFactor = 0.02f;
2	36	}
	37	else
2	38	{
2	39	gallonsRequestedFactor = 0.03f;
2	40	}
	41	
4	42	finalPrices.PricePerGallon = PricePerGallon + ((locationFactor - rateHistoryFactor + gallonsRequestedFactor
4	43	finalPrices.TotalAmount = quoteinfo.GallonsRequested * finalPrices.PricePerGallon;
	44	
4	45	return finalPrices;
4	46	}
	47	
	48	
	49	}
	50	}