

```
1 using Microsoft.VisualStudio.TestTools.UnitTesting;
2 using Vertigo.DAL;
3 using System;
4 using System.Collections.Generic;
5 using System.Linq;
6 using System.Text;
7 using System.Threading.Tasks;
8 using Sfinx;
9
10 namespace Vertigo.DAL.Tests
11 {
12     [TestClass()]
13     public class UmoneyTests
14     {
15         [TestMethod()]
16         public void SelectTest()
17         {
18             try {
19                 Proxy.DBPrivilege.Privilege = "2";
20                 Umoney t = new Umoney("where u_contract_id=1");
21                 List<Umoney> d = t.Data;
22                 int a = d.Count();
23                 Assert.IsTrue(a > 0);
24             }
25             catch(Exception ex)
26             {
27                 Assert.Fail(ex.Message);
28             }
29         }
30
31         [TestMethod()]
32         public void InsertTest()
33         {
34             try
35             {
36                 Proxy.DBPrivilege.Privilege = "2";
37                 Umoney t = new Umoney();
38                 t.Kind = Umoney.KindEnum.PLN.ToString();
39                 t.Status = "A";
40                 t.Period = Umoney.PeriodEnum.mies.ToString();
41                 t.Description = "jakiś opis";
42                 t.Value = 102.34m;
43                 t.U_Contract_Id = 1;
44                 int a = t.Insert();
45                 t.Id = a;
46                 t.Delete();
47                 Assert.IsTrue(a > 0);
48             }
49             catch (Exception ex)
50             {
51                 Assert.Fail(ex.Message);
52             }
53         }
54
55         [TestMethod()]
56         public void UpdateTest()
```

```
57     {
58         try
59         {
60             Proxy.DBPrivilege.Privilege = "2";
61             Umoney t = new Umoney("where u_contract_id=1");
62             List<Umoney> d = t.Data;
63             t = d[1];
64             t.Value = 4.54m;
65             t.Update();
66             Assert.IsTrue(true);
67         }
68         catch (Exception ex)
69         {
70             Assert.Fail(ex.Message);
71         }
72     }
73
74     [TestMethod()]
75     public void DeleteTest()
76     {
77         try
78         {
79             Proxy.DBPrivilege.Privilege = "2";
80             Umoney t = new Umoney();
81             t.Id = 6;
82             t.Delete();
83             Assert.IsTrue(true);
84         }
85         catch (Exception ex)
86         {
87             Assert.Fail(ex.Message);
88         }
89     }
90
91     [TestMethod()]
92     public void EnumTest()
93     {
94         try
95         {
96             Umoney t = new Umoney();
97             var tt = !System.Enum.GetNames(typeof           ↗
98                 (Umoney.ExtEnum)).Contains("Z");
99             tt = !System.Enum.GetNames(typeof(Umoney.ExtEnum)).Contains           ↗
100                 ("Q");
101             Assert.IsTrue(true);
102         }
103         catch (Exception ex)
104         {
105             Assert.Fail(ex.Message);
106         }
107     }
108 }
```