

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»
Факультет компьютерных наук
Департамент программной инженерии**

СОГЛАСОВАНО
Профессор департамента
программной инженерии факультета
компьютерных наук
_____ В.В. Подбельский
«__» _____ 2020 г.

УТВЕРЖДАЮ
Академический руководитель
образовательной программы
«Программная инженерия»
_____ В.В. Шилов
«__» _____ 2020 г.

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл.	

**Программа для расчета и визуализации бинарных фазовых диаграмм в
системе изоструктурных компонентов**

Текст программы

**ЛИСТ УТВЕРЖДЕНИЯ
RU.17701729.04.15-01 12 01-1-ЛУ**

Исполнитель
Студент группы БПИ173
_____/К.Г. Кожакин/
«__» _____ 2020 г.

Москва 2020

УТВЕРЖДЕН
RU.17701729.04.15-01 12 01-1-ЛУ

**Программа для расчета и визуализации бинарных фазовых диаграмм в системе
изоструктурных компонентов**

Текст программы

RU.17701729.04.15-01 12 01-1

Листов 149

Инв. № подл	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Москва 2020

Оглавление

1. Класс MendeleevTable:	4
• MendeleevTable.xaml	4
• MendeleevTable.cs	7
2. Класс ElemContent:	15
• ElemContent.xaml	15
• ElemContent.cs	17
3. Класс ComposSystemContent:	23
• ComposSystemContent.xaml	24
• ComposSystemContent.cs	26
4. Класс CreateComposSystem:	37
• CreateComposSystem.xaml	37
• CreateComposSystem.cs	38
5. Класс ColumnAddWindow:	40
• ColumnAddWindow.xaml	40
• ColumnAddWindow.cs	40
6. Класс FormulaInput:	42
• FormulaInput.xaml	42
• FormulaInput.cs	44
7. Класс FormulaList:	45
• FormulaList.xaml	45
• FormulaList.cs	46
8. Класс DataSettings:	47
• DataSettings.xaml	47
• DataSettings.cs	56
9. Класс DomeOfDecay:	58
• DomeOfDecay.xaml	59
• DomeOfDecay.cs	65
10.Класс Help:	80
• Help.xaml	80
• Help.cs	80

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

11.Классы Approximation:	86
• Criterion.cs.....	86
• Library.cs	88
12.Классы DrawingClasses:.....	94
• Collapse.cs	94
• CollapseGraph.cs	96
13.Классы HelperClasses:	103
• BinSystem.cs.....	103
• Composition.cs.....	113
• DataGridHelper.cs	114
• HotKey.cs	120
• MathParser.cs.....	124
• Pair.cs.....	140
• Point.cs.....	141
• RoundConverter.cs.....	141
• Strings.cs:.....	142
a. StringHelper.....	142
b. Strings	145
СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ.....	148
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ.....	149

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

1. Класс MendeleevTable:

• MendeleevTable.xaml

```

1. <Window x:Name="MendeleevTable1"
   x:Class="Activision_Mendeleev_table.MendeleevTable"
2.     xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
3.     xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
4.     xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
5.     xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
6.     mc:Ignorable="d"
7.     Icon="icon.jpeg"
8.     Title="Таблица Менделеева" Height="720" Width="1280"
   WindowStartupLocation="CenterScreen" ScrollViewer.CanContentScroll="False"
   Loaded="Window_Loaded" SizeChanged="Window_SizeChanged" MinWidth="1280"
   MinHeight="720" FontSize="18">
9.     <Grid ScrollViewer.CanContentScroll="False">
10.         <Grid.RowDefinitions>
11.             <RowDefinition Height="25" />
12.             <RowDefinition Height="*" />
13.         </Grid.RowDefinitions>
14.         <Menu x:Name="MainMenu" Grid.Row="0" Height="25"
   VerticalAlignment="Top">
15.             <Button Name="Composition" Content="Работа с соединением"
   Click="Composition_Click" />
16.             <Button Name="Compositions" Content="Работа с системой соединений"
   Click="Compositions_Click" />
17.             <Separator Name="sep" Width="850" Opacity="0"/>
18.             <Button Name="Help" Content="Справка" Click="Help_Click"/>
19.         </Menu>
20.         <DataGrid Grid.Row="1" x:Name="table" AutoGenerateColumns="False"
   ColumnWidth="2*" VerticalGridLinesBrush="White"
21.             HorizontalGridLinesBrush="White"
   CanUserReorderColumns="False" CanUserSortColumns="False" CanUserAddRows="False"
22.             CanUserDeleteRows="False" CanUserResizeColumns="False"
   CanUserResizeRows="False" IsReadOnly="True" SelectionUnit="Cell"
23.             SelectionMode="Single" GridLinesVisibility="None"
   ScrollViewer.CanContentScroll="False">
24.             <DataGrid.Columns>
25.                 <DataGridTextColumn Header="" Width="1*" />
26.                 <DataGridTextColumn Header="Ia" Binding="{Binding Ia}" />
27.                 <DataGridTextColumn Header="IIa" Binding="{Binding IIa}" />
28.                 <DataGridTextColumn Header="IIIa" Binding="{Binding IIIa}" />

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

29.      <DataGridTextColumn Header="IVa" Binding="{Binding IVa}"/>
30.      <DataGridTextColumn Header="Va" Binding="{Binding Va}"/>
31.      <DataGridTextColumn Header="VIa" Binding="{Binding VIa}"/>
32.      <DataGridTextColumn Header="VIIa" Binding="{Binding VIIa}"/>
33.      <DataGridTextColumn Header="" Binding="{Binding VIIa_1}"/>
34.      <DataGridTextColumn Header="VIIa" Binding="{Binding
      VIIIa_2}"/>
35.      <DataGridTextColumn Header="" Binding="{Binding VIIa_3}"/>
36.      <DataGridTextColumn Header="Iб" Binding="{Binding Ib}"/>
37.      <DataGridTextColumn Header="IIб" Binding="{Binding IIb}"/>
38.      <DataGridTextColumn Header="IIIб" Binding="{Binding IIIb}"/>
39.      <DataGridTextColumn Header="IVб" Binding="{Binding IVb}"/>
40.      <DataGridTextColumn Header="Vб" Binding="{Binding Vb}"/>
41.      <DataGridTextColumn Header="VIб" Binding="{Binding VIb}"/>
42.      <DataGridTextColumn Header="VIIб" Binding="{Binding VIIb}"/>
43.      <DataGridTextColumn Header="VIIIб" Binding="{Binding VIIIb}"/>
44.      <DataGridTextColumn Header="" Width="1*"/>
45.  </DataGrid.Columns>
46.  <DataGrid.ColumnHeaderStyle>
47.      <Style TargetType="{x:Type DataGridColumnHeader}">
48.          <Setter Property="SeparatorBrush" Value="Transparent"/>
49.          <Setter Property="Thumb.HorizontalContentAlignment"
      Value="Stretch"/>
50.          <Setter Property="Thumb.VerticalContentAlignment"
      Value="Stretch"/>
51.      </Style>
52.  </DataGrid.ColumnHeaderStyle>
53.  <DataGrid.CellStyle>
54.      <Style TargetType="DataGridCell">
55.          <Setter Property="FrameworkElement.HorizontalAlignment"
      Value="Stretch"/>
56.          <Setter Property="FrameworkElement.VerticalAlignment"
      Value="Stretch"/>
57.          <EventSetter Event="MouseDoubleClick"
      Handler="DataGridCell_MouseDoubleClick"/>
58.          <Style.Triggers>
59.              <Trigger Property="IsSelected" Value="True">
60.                  <Setter Property="Background" Value="Transparent"/>
61.                  <Setter Property="Foreground" Value="Black" />
62.                  <Setter Property="BorderThickness" Value="0"/>
63.              </Trigger>
64.          </Style.Triggers>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

65.         </Style>
66.     </DataGrid.CellStyle>
67.     <DataGrid.RowHeaderStyle>
68.         <Style TargetType="{x:Type DataGridRowHeader}">
69.             <Setter Property="Content" Value=""/>
70.             <Setter Property="Width" Value="0"/>
71.         </Style>
72.     </DataGrid.RowHeaderStyle>
73. </DataGrid>
74. <Label Name="M" Content="М-Металлы" Foreground="HotPink" Grid.Row="1"
    Width="200" Height="35" RenderTransformOrigin="0.5,0.5" >
75.     <Label.RenderTransform>
76.         <TransformGroup>
77.             <ScaleTransform/>
78.             <SkewTransform/>
79.             <RotateTransform/>
80.             <TranslateTransform X="-200" Y="-160"/>
81.         </TransformGroup>
82.     </Label.RenderTransform>
83. </Label>
84. <Label Name="T" Content="Т-Металлы" Foreground="CornflowerBlue"
    Grid.Row="1" Width="200" Height="35" RenderTransformOrigin="0.5,0.5" >
85.     <Label.RenderTransform>
86.         <TransformGroup>
87.             <ScaleTransform/>
88.             <SkewTransform/>
89.             <RotateTransform/>
90.             <TranslateTransform X="-200" Y="160"/>
91.         </TransformGroup>
92.     </Label.RenderTransform>
93. </Label>
94. <Label Name="B1" Content="B1-Металлы" Grid.Row="1" Width="200"
    Height="35" RenderTransformOrigin="0.5,0.5" >
95.     <Label.RenderTransform>
96.         <TransformGroup>
97.             <ScaleTransform/>
98.             <SkewTransform/>
99.             <RotateTransform/>
100.            <TranslateTransform Y="160" X="230"/>
101.        </TransformGroup>
102.    </Label.RenderTransform>
103. </Label>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

104.      <Label Name="B2" Content="B2-Металлы" Grid.Row="1" Width="200"
      Height="35" RenderTransformOrigin="0.5,0.5" >
105.          <Label.RenderTransform>
106.              <TransformGroup>
107.                  <ScaleTransform/>
108.                  <SkewTransform/>
109.                  <RotateTransform/>
110.                  <TranslateTransform Y="160" X="477"/>
111.              </TransformGroup>
112.          </Label.RenderTransform>
113.      </Label>
114.      <Label Name="R" Content="R-Лантаноиды" Grid.Row="1" Width="200"
      Height="35" RenderTransformOrigin="0.5,0.5" >
115.          <Label.RenderTransform>
116.              <TransformGroup>
117.                  <ScaleTransform/>
118.                  <SkewTransform/>
119.                  <RotateTransform/>
120.                  <TranslateTransform Y="215" X="470"/>
121.              </TransformGroup>
122.          </Label.RenderTransform>
123.      </Label>
124.      <Label Name="TR" Content="TR-Актиноиды" Grid.Row="1" Width="200"
      Height="35" RenderTransformOrigin="0.5,0.5" >
125.          <Label.RenderTransform>
126.              <TransformGroup>
127.                  <ScaleTransform/>
128.                  <SkewTransform/>
129.                  <RotateTransform/>
130.                  <TranslateTransform X="470" Y="275"/>
131.              </TransformGroup>
132.          </Label.RenderTransform>
133.      </Label>
134.  </Grid>
135. </Window>

```

- **MendeleevTable.cs**

```

1. using Activision_Mendeleev_table.HelperClasses;
2. using static Activision_Mendeleev_table.HelperClasses.DataGridHelper;
3. using System.Collections.Generic;
4. using System.Windows;

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата


```

5. using System.Windows.Controls;
6. using System.Windows.Input;
7. using System.Windows.Media;
8.
9. namespace Activision_Mendeleev_table
10. {
11.     /// <summary>
12.     /// Логика взаимодействия для MendeleevTable.xaml
13.     /// </summary>
14.     public partial class MendeleevTable : Window
15.     {
16.         /// <summary>
17.         /// Лист элементов
18.         /// </summary>
19.         public static List<Composition> Elems = new List<Composition>();
20.
21.         /// <summary>
22.         /// Лист соединений
23.         /// </summary>
24.         public static List<Composition> Compos = new List<Composition>();
25.
26.         /// <summary>
27.         /// Лист систем соединений
28.         /// </summary>
29.         public static List<System.Data.DataTable> BinarySistem = new
List<System.Data.DataTable>();
30.
31.         /// <summary>
32.         /// Конструктор главного окна
33.         /// </summary>
34.         public MendeleevTable()
35.         {
36.             InitializeComponent();
37.             //Привязка горячей клавише F1 к методу OnF1Handler
38.             new HotKey(Key.F1, KeyModifier.None, OnF1Handler);
39.
40.             List<Strings> strings = new List<Strings>(10)
41.             {
42.                 new Strings(),
43.                 new Strings("Li", "Be"),
44.                 new Strings("Na", "Mg"),

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

45.         new Strings("K", "Ca", "Sc", "Ti", "V", "Cr", "Mn", "Fe", "Co",
    "Ni", "Cu", "Zn", "Ga", "Ge", "As", "Se", "Br", "Kr"),
46.         new Strings("Rb", "Sr", "Y", "Zr", "Nb", "Mo", "Tc", "Ru",
    "Rh", "Pd", "Ag", "Cd", "In", "Sn", "Sb", "Te", "I", "Xe"),
47.         new Strings("Cs", "Ba", "*La", "Hf", "Ta", "W", "Re", "Os",
    "Ir", "Pt", "Au", "Hg", "Tl", "Pb", "Bi", "Po", "At", "Rn"),
48.         new Strings("Fr", "Ra", "**Ac"),
49.         new Strings(),
50.         new Strings("*Ce", "Pr", "Nd", "Pm", "Sm", "Eu", "Gd", "Tb",
    "Dy", "Ho", "Er", "Tm", "Yb", "Lu"),
51.         new Strings("***", "Th", "Pa", "U", "Np", "Pu", "Am", "Cm",
    "Bk", "Cf", "Es", "Fm", "Md", "-")
52.     };
53.
54.     strings[0].VIIb = "H";
55.     strings[0].VIIIb = "He";
56.     strings[1].IIIb = "B";
57.     strings[1].IVb = "C";
58.     strings[1].Vb = "N";
59.     strings[1].VIb = "O";
60.     strings[1].VIIb = "F";
61.     strings[1].VIIIb = "Ne";
62.     strings[2].IIIb = "Al";
63.     strings[2].IVb = "Si";
64.     strings[2].Vb = "P";
65.     strings[2].VIb = "S";
66.     strings[2].VIIb = "Cl";
67.     strings[2].VIIIb = "Ar";
68.     strings[6].VIIIb = "I раз";
69.
70.     table.Items.Clear();
71.     table.ItemsSource = strings;
72.
73.     Deserialize("BinarySystems.xml", ref BinarySystem);
74.     Deserialize("Compositions.xml", ref Compos);
75.
76.     if (!Deserialize("Elements.xml", ref Elems))
77.     {
78.         Elems.Add(new Composition("H"));
79.         Elems.Add(new Composition("Li"));
80.         Elems.Add(new Composition("Be"));
81.         Elems.Add(new Composition("Na"));

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

10
RU.17701729.04.15-01 12 01-1

```

82.      Elems.Add(new Composition("Mg"));
83.      Elems.Add(new Composition("K"));
84.      Elems.Add(new Composition("Ca"));
85.      Elems.Add(new Composition("Sc"));
86.      Elems.Add(new Composition("Ti"));
87.      Elems.Add(new Composition("V"));
88.      Elems.Add(new Composition("Cr"));
89.      Elems.Add(new Composition("Mn"));
90.      Elems.Add(new Composition("Fe"));
91.      Elems.Add(new Composition("Co"));
92.      Elems.Add(new Composition("Ni"));
93.      Elems.Add(new Composition("Cu"));
94.      Elems.Add(new Composition("Zn"));
95.      Elems.Add(new Composition("Ga"));
96.      Elems.Add(new Composition("Ge"));
97.      Elems.Add(new Composition("As"));
98.      Elems.Add(new Composition("Se"));
99.      Elems.Add(new Composition("Br"));
100.     Elems.Add(new Composition("Kr"));
101.     Elems.Add(new Composition("Rb"));
102.     Elems.Add(new Composition("Sr"));
103.     Elems.Add(new Composition("Y"));
104.     Elems.Add(new Composition("Zr"));
105.     Elems.Add(new Composition("Nb"));
106.     Elems.Add(new Composition("Mo"));
107.     Elems.Add(new Composition("Tc"));
108.     Elems.Add(new Composition("Ru"));
109.     Elems.Add(new Composition("**Ac"));
110.     Elems.Add(new Composition("Rh"));
111.     Elems.Add(new Composition("Pd"));
112.     Elems.Add(new Composition("Ag"));
113.     Elems.Add(new Composition("Cd"));
114.     Elems.Add(new Composition("In"));
115.     Elems.Add(new Composition("Sn"));
116.     Elems.Add(new Composition("Sb"));
117.     Elems.Add(new Composition("Te"));
118.     Elems.Add(new Composition("I"));
119.     Elems.Add(new Composition("Xe"));
120.     Elems.Add(new Composition("Cs"));
121.     Elems.Add(new Composition("Ba"));
122.     Elems.Add(new Composition("Hf"));
123.     Elems.Add(new Composition("Ta"));

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

11
RU.17701729.04.15-01 12 01-1

```

124.      Elems.Add(new Composition("W"));
125.      Elems.Add(new Composition("Re"));
126.      Elems.Add(new Composition("Os"));
127.      Elems.Add(new Composition("Ir"));
128.      Elems.Add(new Composition("Pt"));
129.      Elems.Add(new Composition("Au"));
130.      Elems.Add(new Composition("Hg"));
131.      Elems.Add(new Composition("Tl"));
132.      Elems.Add(new Composition("Pb"));
133.      Elems.Add(new Composition("Bi"));
134.      Elems.Add(new Composition("Po"));
135.      Elems.Add(new Composition("At"));
136.      Elems.Add(new Composition("Rn"));
137.      Elems.Add(new Composition("Fr"));
138.      Elems.Add(new Composition("Ra"));
139.      Elems.Add(new Composition("He"));
140.      Elems.Add(new Composition("B"));
141.      Elems.Add(new Composition("C"));
142.      Elems.Add(new Composition("N"));
143.      Elems.Add(new Composition("O"));
144.      Elems.Add(new Composition("F"));
145.      Elems.Add(new Composition("Ne"));
146.      Elems.Add(new Composition("Al"));
147.      Elems.Add(new Composition("Si"));
148.      Elems.Add(new Composition("P"));
149.      Elems.Add(new Composition("S"));
150.      Elems.Add(new Composition("Cl"));
151.      Elems.Add(new Composition("Ar"));
152.      Elems.Add(new Composition("I раз"));
153.      Elems.Add(new Composition("*Ce"));
154.      Elems.Add(new Composition("Pr"));
155.      Elems.Add(new Composition("Nd"));
156.      Elems.Add(new Composition("Pm"));
157.      Elems.Add(new Composition("Sm"));
158.      Elems.Add(new Composition("Eu"));
159.      Elems.Add(new Composition("Gd"));
160.      Elems.Add(new Composition("Tb"));
161.      Elems.Add(new Composition("Dy"));
162.      Elems.Add(new Composition("Ho"));
163.      Elems.Add(new Composition("Er"));
164.      Elems.Add(new Composition("Tm"));
165.      Elems.Add(new Composition("Yb"));

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

12
RU.17701729.04.15-01 12 01-1

```
166.         Elems.Add(new Composition("Lu"));
167.         Elems.Add(new Composition("Th"));
168.         Elems.Add(new Composition("Pa"));
169.         Elems.Add(new Composition("U"));
170.         Elems.Add(new Composition("Np"));
171.         Elems.Add(new Composition("Pu"));
172.         Elems.Add(new Composition("Am"));
173.         Elems.Add(new Composition("Cm"));
174.         Elems.Add(new Composition("Bk"));
175.         Elems.Add(new Composition("Cf"));
176.         Elems.Add(new Composition("Es"));
177.         Elems.Add(new Composition("Fm"));
178.         Elems.Add(new Composition("Md"));
179.         Elems.Add(new Composition("*La"));
180.
181.     }
182. }
183.
184.     /// <summary>
185.     /// Открывает окно работы с системой соединений
186.     /// </summary>
187.     private void Compositions_Click(object sender, RoutedEventArgs e)
188.     {
189.         new CreateComposSystem(false).ShowDialog();
190.     }
191.
192.     /// <summary>
193.     /// Открывает окно работы с соединением
194.     /// </summary>
195.     private void Composition_Click(object sender, RoutedEventArgs e)
196.     {
197.         new CreateComposSystem(true).ShowDialog();
198.     }
199.
200.     private void Window_Loaded(object sender, RoutedEventArgs e)
201.     {
202.         for (int i = 3; i < 12; i++)
203.             for (int j = 3; j < 7; j++)
204.                 BrushCell(j, i, Brushes.LightBlue, table);
205.
206.         for (int i = 1; i <= 2; i++)
207.             for (int j = 1; j < 7; j++)
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

13
RU.17701729.04.15-01 12 01-1

```
208.         BrushCell(j, i, Brushes.LightPink, table);
209.     BrushCell(2, 13, Brushes.LightPink, table);
210.
211.     for (int i = 0; i <= 6; i++)
212.         BrushCell(i, 18, Brushes.Orange, table);
213.
214.     for (int i = 12; i <= 14; i++)
215.         for (int j = 3; j < 7; j++)
216.             BrushCell(j, i, Brushes.AntiqueWhite, table);
217.     BrushCell(3, 14, Brushes.White, table);
218.
219.     BrushCell(1, 13, Brushes.LightGreen, table);
220.     BrushCell(2, 14, Brushes.LightGreen, table);
221.     BrushCell(3, 15, Brushes.LightGreen, table);
222.     BrushCell(4, 16, Brushes.LightGreen, table);
223.     BrushCell(5, 17, Brushes.LightGreen, table);
224.
225.     }
226.
227.     private void Window_SizeChanged(object sender,
        SizeChangedEventArgs e)
228.     {
229.         DataGridViewCell cell;
230.         double height = e.NewSize.Height;
231.         double width = e.NewSize.Width;
232.         sep.Width = 850 + (width - 1280);
233.         for (int i = 0; i <= 19; i++)
234.         {
235.             for (int j = 0; j < 10; j++)
236.             {
237.                 cell = GetCell(j, i, table);
238.                 cell.Height = 68 + (height - 780) / 10;
239.                 cell.FontSize = 20 + (height - 780) / 100;
240.             }
241.         }
242.         M.FontSize = 18 + (height - 780) / 100;
243.         T.FontSize = 18 + (height - 780) / 100;
244.         B1.FontSize = 18 + (height - 780) / 100;
245.         B2.FontSize = 18 + (height - 780) / 100;
246.         R.FontSize = 18 + (height - 780) / 100;
247.         TR.FontSize = 18 + (height - 780) / 100;
248.         TranslateTransform transform = new TranslateTransform
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

14
RU.17701729.04.15-01 12 01-1

```
249.      {
250.          X = -180 - (width - 1280) / 4.5,
251.          Y = -170 - (height - 780) / 4.5
252.      };
253.      M.RenderTransform = transform;
254.      transform = new TranslateTransform
255.      {
256.          X = -180 - (width - 1280) / 4.5,
257.          Y = 180 + (height - 780) / 4.5
258.      };
259.      T.RenderTransform = transform;
260.      transform = new TranslateTransform
261.      {
262.          X = 260 + (width - 1280) / 6,
263.          Y = 180 + (height - 780) / 4.5
264.      };
265.      B1.RenderTransform = transform;
266.      transform = new TranslateTransform
267.      {
268.          X = 500 + (width - 1280) / 2.5,
269.          Y = 180 + (height - 780) / 4.5
270.      };
271.      B2.RenderTransform = transform;
272.      transform = new TranslateTransform
273.      {
274.          X = 490 + (width - 1280) / 2.5,
275.          Y = 230 + (height - 780) / 3
276.      };
277.      R.RenderTransform = transform;
278.      transform = new TranslateTransform
279.      {
280.          X = 490 + (width - 1280) / 2.5,
281.          Y = 295 + (height - 780) / 2.5
282.      };
283.      TR.RenderTransform = transform;
284.      }
285.
286.      /// <summary>
287.      /// Открывает окно работы с элементом
288.      /// </summary>
289.      private void DataGridCell_MouseDoubleClick(object sender,
        MouseButtonEventArgs e)
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```
290.      {
291.          var dataGridCellTarget = (DataGridCell)sender;
292.          if (dataGridCellTarget != null)
293.          {
294.              TextBlock text = dataGridCellTarget.Content as TextBlock;
295.              if (text.Text != "" && text.Text != "-" && text.Text !=
                "***")
296.                  new ElemContent(text.Text).ShowDialog();
297.          }
298.      }
299.
300.      /// <summary>
301.      /// Открывает окно справки
302.      /// </summary>
303.      private void Help_Click(object sender, RoutedEventArgs e)
304.      {
305.          new Help().Show();
306.      }
307.
308.      /// <summary>
309.      /// Открывает окно справки
310.      /// </summary>
311.      /// <param name="hotKey">горячая клавиша для вызова
        метода</param>
312.      private static void OnF1Handler(HotKey hotKey)
313.      {
314.          new Help().Show();
315.      }
316.  }
317. }
```

2. Класс ElemContent:

- ElemContent.xaml

```
1. <Window
2.     xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
3.     xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
4.     xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
5.     xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
6.     x:Class="Activision_Mendeleyev_table.ElemContent"
7.     mc:Ignorable="d"
8.     Icon="icon.jpeg"
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

16
RU.17701729.04.15-01 12 01-1

```
9.      Title="ElemContent" Height="450" Width="880" Closing="Window_Closing"
WindowStartupLocation="CenterScreen" MinHeight="450" MinWidth="880"
SizeChanged="Window_SizeChanged">
10.      <Grid>
11.          <Grid.RowDefinitions>
12.              <RowDefinition Height="*" />
13.              <RowDefinition Height="50"/>
14.          </Grid.RowDefinitions>
15.          <DataGrid x:Name="ElemTable" Grid.Row="0" CanUserSortColumns="False"
CanUserReorderColumns="False" CanUserResizeColumns="False" ColumnWidth="*"
16.              AutoGenerateColumns="False" CanUserAddRows="False"
CanUserDeleteRows="False" CanUserResizeRows="False" IsReadOnly="True"
CellEditEnding="ElemContent_CellEditEnding"/>
17.          <Button Name="AddColumn" Grid.Row="1" Content="Добавить столбец"
Click="AddColumn_Click" Width="150" Height="40" HorizontalAlignment="Left"
RenderTransformOrigin="0.5,0.5" Visibility="Hidden">
18.              <Button.RenderTransform>
19.                  <TransformGroup>
20.                      <ScaleTransform/>
21.                      <SkewTransform/>
22.                      <RotateTransform/>
23.                      <TranslateTransform X="20"/>
24.                  </TransformGroup>
25.              </Button.RenderTransform>
26.          </Button>
27.          <Button Name="AddRow" Grid.Row="1" Content="Добавить строку"
Click="AddRow_Click" Width="150" Height="40" RenderTransformOrigin="0.5,0.5"
HorizontalAlignment="Left" Visibility="Hidden">
28.              <Button.RenderTransform>
29.                  <TransformGroup>
30.                      <ScaleTransform/>
31.                      <SkewTransform/>
32.                      <RotateTransform/>
33.                      <TranslateTransform X="190"/>
34.                  </TransformGroup>
35.              </Button.RenderTransform>
36.          </Button>
37.          <Button Name="DelColumn" Grid.Row="1" Content="Удалить столбец"
Click="DelColumn_Click" Width="150" Height="40" HorizontalAlignment="Left"
RenderTransformOrigin="0.5,0.5" Visibility="Hidden" IsEnabled="False">
38.              <Button.RenderTransform>
39.                  <TransformGroup>
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

40.             <ScaleTransform/>
41.             <SkewTransform/>
42.             <RotateTransform/>
43.             <TranslateTransform X="360"/>
44.         </TransformGroup>
45.     </Button.RenderTransform>
46. </Button>
47.     <Button Name="DelSelectedRows" Grid.Row="1" Content="Удалить выделенные
строки" Click="DelSelectedRows_Click" Width="150" Height="40"
HorizontalAlignment="Left" RenderTransformOrigin="0.5,0.5" Visibility="Hidden"
IsEnabled="False" FontSize="10">
48.         <Button.RenderTransform>
49.             <TransformGroup>
50.                 <ScaleTransform/>
51.                 <SkewTransform/>
52.                 <RotateTransform/>
53.                 <TranslateTransform X="530"/>
54.             </TransformGroup>
55.         </Button.RenderTransform>
56.     </Button>
57.     <Button Name="Save" Grid.Row="1" Content="Сохранить изменения"
Click="Save_Click" Width="150" Height="40" HorizontalAlignment="Left"
RenderTransformOrigin="0.5,0.5" Visibility="Hidden">
58.         <Button.RenderTransform>
59.             <TransformGroup>
60.                 <ScaleTransform/>
61.                 <SkewTransform/>
62.                 <RotateTransform/>
63.                 <TranslateTransform X="700"/>
64.             </TransformGroup>
65.         </Button.RenderTransform>
66.     </Button>
67.     <Button Name="EditTable" Grid.Row="1" Content="Редактировать таблицу"
Click="EditTable_Click" Width="150" Height="40"/>
68. </Grid>
69. </Window>

```

• **ElemContent.cs**

```

1. using System.Collections.Generic;
2. using System.Data;
3. using System.Windows;

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```
4. using System.Windows.Controls;
5. using System.Windows.Data;
6. using System.Windows.Media;
7. using Activision_Mendeleyev_table.HelperClasses;
8.
9. namespace Activision_Mendeleyev_table
10. {
11.     /// <summary>
12.     /// Логика взаимодействия для ElemContent.xaml
13.     /// </summary>
14.     public partial class ElemContent : Window
15.     {
16.         /// <summary>
17.         /// Таблица данных
18.         /// </summary>
19.         private DataTable dat;
20.
21.         /// <summary>
22.         /// Обозначение элемента
23.         /// </summary>
24.         private string elem;
25.
26.         /// <summary>
27.         /// Конструктор, инициализирующий окно таблицы элемента
28.         /// </summary>
29.         /// <param name="elem">название элемента</param>
30.         public ElemContent(string elem)
31.         {
32.             InitializeComponent();
33.             this.Title = "Свойства элемента " + elem;
34.             this.elem = elem;
35.
36.             Composition comp = MendeleevTable.Elems.Find(x => x.Name == elem);
37.             if (comp != null)
38.             {
39.                 //Заполнение строк
40.                 dat = comp.DataTable;
41.                 for (int i = 0; i < comp.Properties.Count; i++)
42.                     for (int j = 0; j < dat.Columns.Count; j++)
43.                         if (dat.Columns[j].ColumnName ==
                             comp.Properties[i].First.First)
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

19
RU.17701729.04.15-01 12 01-1

```

44.         for (int k = 0; k <
comp.Properties[i].Second.Count; k++)
45.         {
46.             if (dat.Rows.Count <= k)
47.                 dat.Rows.Add();
48.             dat.Rows[k][j] = comp.Properties[i].Second[k];
49.         }
50.     }
51.
52.     if (dat == null)
53.         dat = new DataTable() { TableName = elem };
54.     else
55.     {
56.         //Визуализация столбцов
57.         foreach (DataColumn i in dat.Columns)
58.         {
59.             ElemTable.Columns.Add(new DataGridTextColumn()
60.             {
61.                 Header = (i.Caption == "" || i.Caption == " ") ?
i.ColumnName : i.ColumnName + ", " + i.Caption,
62.                 Binding = new Binding "[" + ElemTable.Columns.Count +
 "]"")
63.             });
64.         }
65.     }
66.
67.     ElemTable.ItemsSource = dat.DefaultView;
68.     if (dat.Columns.Count > 0)
69.         DelColumn.IsEnabled = true;
70.     if (dat.Rows.Count > 0)
71.         DelSelectedRows.IsEnabled = true;
72. }
73.
74.     /// <summary>
75.     /// Добавляет текстовый столбец в таблицу
76.     /// </summary>
77.     private void AddColumn_Click(object sender, RoutedEventArgs e)
78.     {
79.         DelColumn.IsEnabled = DataGridHelper.AddColumn(ref ElemTable, ref
dat);
80.     }
81.

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

82.         /// <summary>
83.         /// Добавляет строку в таблицу
84.         /// </summary>
85.         private void AddRow_Click(object sender, RoutedEventArgs e)
86.         {
87.             dat.Rows.Add();
88.             if (dat.Rows.Count > 0)
89.                 DelSelectedRows.IsEnabled = true;
90.
91.             CollectionViewSource.GetDefaultView(ElemTable.ItemsSource).Refresh();
92.         }
93.
94.         /// <summary>
95.         /// Удаляет столбец в таблицу
96.         /// </summary>
97.         private void DelColumn_Click(object sender, RoutedEventArgs e)
98.         {
99.             ElemTable.Columns.RemoveAt(ElemTable.Columns.Count - 1);
100.            if (dat.Columns.Count <= 1)
101.                DelColumn.IsEnabled = false;
102.            dat.Columns.RemoveAt(dat.Columns.Count - 1);
103.        }
104.
105.        /// <summary>
106.        /// Удаляет строку в таблицу
107.        /// </summary>
108.        private void DelSelectedRows_Click(object sender, RoutedEventArgs
109.        e)
110.        {
111.            try
112.            {
113.                while (ElemTable.SelectedItems.Count > 0)
114.                {
115.                    int selectedIndex = ElemTable.SelectedIndex;
116.
117.                    DrawingClasses.CollapseGraph.RemoveSelectedPoint(selectedIndex);
118.                    dat.Rows.RemoveAt(selectedIndex);
119.                    ElemTable.Items.Refresh();
120.                }
121.            }
122.            catch (System.Exception)

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

121.         {
122.             MessageBox.Show("Невозможно удалить этот элемент!",
123.                 "Error", MessageBoxButton.OK, MessageBoxImage.Error);
124.         }
125.         if (dat.Rows.Count == 0)
126.             DelSelectedRows.IsEnabled = false;
127.     }
128.     /// <summary>
129.     /// Сохраняет данные в файл
130.     /// </summary>
131.     private void Save_Click(object sender, RoutedEventArgs e)
132.     {
133.         ElemTable.IsReadOnly = true;
134.         EditTable.Visibility = Visibility.Visible;
135.         AddColumn.Visibility = Visibility.Hidden;
136.         AddRow.Visibility = Visibility.Hidden;
137.         DelColumn.Visibility = Visibility.Hidden;
138.         DelSelectedRows.Visibility = Visibility.Hidden;
139.         Save.Visibility = Visibility.Hidden;
140.
141.         dat.AcceptChanges();
142.         List<Pair<Pair<string, string>, List<string>>>> prop = new
143.             List<Pair<Pair<string, string>, List<string>>>>();
144.
145.         if (prop == null)
146.             prop = new List<Pair<Pair<string, string>,
147.                 List<string>>>>();
148.
149.         for (int i = 0; i < dat.Columns.Count; i++)
150.         {
151.             prop.Add(new Pair<Pair<string, string>, List<string>>>(new
152.                 Pair<string, string>(dat.Columns[i].ColumnName, dat.Columns[i].Caption), new
153.                 List<string>()));
154.             for (int j = 0; j < dat.Rows.Count; j++)
155.                 prop[i].Second.Add(dat.Rows[j][i].ToString());
156.         }
157.
158.         Composition el = MendeleevTable.Elems.Find(x => x.Name ==
159.             elem);
160.
161.         if (el != null)

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

156.                MendeleevTable.Elems.Find(x => x.Name == elem).Properties
    = new List<Pair<Pair<string, string>, List<string>>>();
157.                else
158.                MendeleevTable.Elems.Add(new Composition() { Name = elem
    });
159.
160.                MendeleevTable.Elems.Find(x => x.Name == elem).Properties =
    prop;
161.
162.                DataTable d = dat.Copy();
163.                d.Rows.Clear();
164.                for (int i = 0; i < MendeleevTable.Elems.Count; i++)
165.                    MendeleevTable.Elems[i].DataTable = d;
166.
167.                DataGridHelper.Serialize("Elems.xml", ref
    MendeleevTable.Elems);
168.            }
169.
170.            private void ElemContent_CellEditEnding(object sender,
    DataGridCellEditEndingEventArgs e)
171.            {
172.                string str1 = StringHelper.DoString((e.EditingElement as
    TextBox).Text);
173.                str1 = str1.Replace('.', ',');
174.                dat.Rows[e.Row.GetIndex()][e.Column.DisplayIndex] = str1;
175.                (e.EditingElement as TextBox).Text = str1;
176.            }
177.
178.            private void Window_Closing(object sender,
    System.ComponentModel.CancelEventArgs e)
179.            {
180.                if (EditTable.Visibility == Visibility.Hidden)
181.                    if (MessageBox.Show("Вы точно хотите закрыть окно? Все
    несохраненные данные будут удалены!", "", MessageBoxButton.YesNo,
    MessageBoxImage.Question) == MessageBoxResult.No)
182.                        e.Cancel = true;
183.                    else
184.                        dat.RejectChanges();
185.            }
186.
187.            /// <summary>
188.            /// Позволяет редактировать таблицу

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

189.          /// </summary>
190.          private void EditTable_Click(object sender, RoutedEventArgs e)
191.          {
192.              ElemTable.IsReadOnly = false;
193.              EditTable.Visibility = Visibility.Hidden;
194.              AddColumn.Visibility = Visibility.Visible;
195.              AddRow.Visibility = Visibility.Visible;
196.              DelColumn.Visibility = Visibility.Visible;
197.              DelSelectedRows.Visibility = Visibility.Visible;
198.              Save.Visibility = Visibility.Visible;
199.          }
200.
201.          private void Window_SizeChanged(object sender,
202.              SizeChangedEventArgs e)
203.          {
204.              if (e.PreviousSize.Width != 0)
205.              {
206.                  AddRow.Width += (e.NewSize.Width - e.PreviousSize.Width)
207.                  / 5;
208.                  AddColumn.Width += (e.NewSize.Width -
209.                  e.PreviousSize.Width) / 5;
210.                  DelColumn.Width += (e.NewSize.Width -
211.                  e.PreviousSize.Width) / 5;
212.                  DelSelectedRows.Width += (e.NewSize.Width -
213.                  e.PreviousSize.Width) / 5;
214.                  Save.Width += (e.NewSize.Width - e.PreviousSize.Width) /
215.                  5;
216.                  DelColumn.RenderTransform = new TranslateTransform(360 +
217.                  (e.NewSize.Width - 880) / 2.5, 0);
218.                  AddRow.RenderTransform = new TranslateTransform(190 +
219.                  (e.NewSize.Width - 880) / 5, 0);
220.                  Save.RenderTransform = new TranslateTransform(700 +
221.                  (e.NewSize.Width - 880) / 1.25, 0);
222.                  DelSelectedRows.RenderTransform = new
223.                  TranslateTransform(530 + (e.NewSize.Width - 880) / 1.66, 0);
224.              }
225.          }
226.      }
227.  }
228.

```

3. Класс ComposSystemContent:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

- **ComposSystemContent.xaml**

```

1. <Window x:Class="Activision_Mendeleyev_table.ComposSystemContent"
2.     xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
3.     xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
4.     xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
5.     xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
6.     mc:Ignorable="d"
7.     Icon="icon.jpeg"
8.     Title="ComposSystemContent" Height="450" Width="1050"
   WindowStartupLocation="CenterScreen" MinHeight="450" MinWidth="1050"
   Closing="Window_Closing" SizeChanged="Window_SizeChanged">
9.     <Grid>
10.         <Grid.RowDefinitions>
11.             <RowDefinition Height="*" />
12.             <RowDefinition Height="50"/>
13.         </Grid.RowDefinitions>
14.         <DataGrid x:Name="ComposSystemTable" Grid.Row="0" IsReadOnly="True"
   CanUserSortColumns="False" CanUserReorderColumns="False"
   CanUserResizeColumns="False" ColumnWidth="*"
15.             AutoGenerateColumns="False" CanUserAddRows="False"
   CanUserDeleteRows="False" CanUserResizeRows="False"
   CellEditEnding="ComposSystemTable_CellEditEnding"/>
16.         <Button Name="AddFormul" Grid.Row="1" Visibility="Hidden"
   Content="Добавить формулу" Click="AddFormul_Click" Width="150" Height="40"
   HorizontalAlignment="Left" RenderTransformOrigin="0.5,0.5">
17.             <Button.RenderTransform>
18.                 <TransformGroup>
19.                     <ScaleTransform/>
20.                     <SkewTransform/>
21.                     <RotateTransform/>
22.                     <TranslateTransform X="20"/>
23.                 </TransformGroup>
24.             </Button.RenderTransform>
25.         </Button>
26.         <Button Name="AddColumn" Grid.Row="1" Visibility="Hidden"
   Content="Добавить столбец" Click="AddColumn_Click" Width="150" Height="40"
   HorizontalAlignment="Left" RenderTransformOrigin="0.5,0.5">
27.             <Button.RenderTransform>
28.                 <TransformGroup>
29.                     <ScaleTransform/>
30.                     <SkewTransform/>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

31.             <RotateTransform/>
32.             <TranslateTransform X="190"/>
33.         </TransformGroup>
34.     </Button.RenderTransform>
35. </Button>
36.     <Button Name="AddRow" Grid.Row="1" Visibility="Hidden"
Content="Добавить строку" Click="AddRow_Click" Width="150" Height="40"
RenderTransformOrigin="0.5,0.5" HorizontalAlignment="Left">
37.         <Button.RenderTransform>
38.             <TransformGroup>
39.                 <ScaleTransform/>
40.                 <SkewTransform/>
41.                 <RotateTransform/>
42.                 <TranslateTransform X="360"/>
43.             </TransformGroup>
44.         </Button.RenderTransform>
45.     </Button>
46.     <Button Name="DelColumn" Grid.Row="1" Visibility="Hidden"
Content="Удалить столбец" Click="DelColumn_Click" Width="150" Height="40"
HorizontalAlignment="Left" RenderTransformOrigin="0.5,0.5" IsEnabled="False">
47.         <Button.RenderTransform>
48.             <TransformGroup>
49.                 <ScaleTransform/>
50.                 <SkewTransform/>
51.                 <RotateTransform/>
52.                 <TranslateTransform X="530"/>
53.             </TransformGroup>
54.         </Button.RenderTransform>
55.     </Button>
56.     <Button Name="DelSelectedRows" Grid.Row="1" Visibility="Hidden"
FontSize="10" Content="Удалить выделенные строки" Click="DelSelectedRows_Click"
Width="150" Height="40" HorizontalAlignment="Left"
RenderTransformOrigin="0.5,0.5" IsEnabled="False">
57.         <Button.RenderTransform>
58.             <TransformGroup>
59.                 <ScaleTransform/>
60.                 <SkewTransform/>
61.                 <RotateTransform/>
62.                 <TranslateTransform X="700"/>
63.             </TransformGroup>
64.         </Button.RenderTransform>
65.     </Button>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

66.         <Button Name="Calculate" Grid.Row="1" Visibility="Hidden"
Content="Рассчитать и сохранить" Click="Calculate_Click" Width="150"
Height="40" HorizontalAlignment="Left" RenderTransformOrigin="0.5,0.5">
67.             <Button.RenderTransform>
68.                 <TransformGroup>
69.                     <ScaleTransform/>
70.                     <SkewTransform/>
71.                     <RotateTransform/>
72.                     <TranslateTransform X="870"/>
73.                 </TransformGroup>
74.             </Button.RenderTransform>
75.         </Button>
76.         <Button Name="EditTable" Grid.Row="1" Content="Редактировать таблицу"
Click="EditTable_Click" Width="150" Height="40"
RenderTransformOrigin="0.5,0.5" >
77.             <Button.RenderTransform>
78.                 <TransformGroup>
79.                     <ScaleTransform/>
80.                     <SkewTransform/>
81.                     <RotateTransform/>
82.                     <TranslateTransform X="-100"/>
83.                 </TransformGroup>
84.             </Button.RenderTransform>
85.         </Button>
86.         <Button Name="DomeOfDecayWindowOpen" Grid.Row="1" Content="Купол
распада" Click="DomeOfDecayWindowOpen_Click" Width="150" Height="40"
RenderTransformOrigin="0.5,0.5">
87.             <Button.RenderTransform>
88.                 <TransformGroup>
89.                     <ScaleTransform/>
90.                     <SkewTransform/>
91.                     <RotateTransform/>
92.                     <TranslateTransform X="100"/>
93.                 </TransformGroup>
94.             </Button.RenderTransform>
95.         </Button>
96.     </Grid>
97.</Window>

```

- **ComposSystemContent.cs**

```
1. using Activision_Mendeleyev_table.HelperClasses;
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

2. using System;
3. using System.Collections.Generic;
4. using System.Data;
5. using System.Windows;
6. using System.Windows.Controls;
7. using System.Windows.Data;
8. using System.Windows.Media;
9.
10. namespace Activision_Mendeleyev_table
11. {
12.     /// <summary>
13.     /// Логика взаимодействия для ComposSystemContent.xaml
14.     /// </summary>
15.     public partial class ComposSystemContent : Window
16.     {
17.         /// <summary>
18.         /// Таблица данных
19.         /// </summary>
20.         private DataTable dat;
21.
22.         /// <summary>
23.         /// Обозначение соединения(системы соединений)
24.         /// </summary>
25.         private string elem;
26.
27.         /// <summary>
28.         /// Флаг: true - соединение, false - система
29.         /// </summary>
30.         private bool f;
31.
32.         /// <summary>
33.         /// Конструктор, инициализирующий окно таблицы соединения(системы
            соединений)
34.         /// </summary>
35.         /// <param name="elem">название соединения(системы соединений)</param>
36.         /// <param name="f">флаг: true - соединение, false - система</param>
37.         public ComposSystemContent(string elem, bool f)
38.         {
39.             InitializeComponent();
40.             this.f = f;
41.             this.elem = elem;
42.

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

43.         if (f)
44.         {
45.             this.Title = "Таблица соединения " + elem;
46.             DomeOfDecayWindowOpen.Visibility = Visibility.Hidden;
47.             EditTable.RenderTransform = new TranslateTransform();
48.
49.             Composition comp = MendeleevTable.Compos.Find(x => x.Name ==
elem);
50.             dat = new DataTable() { TableName = elem };
51.             if (comp != null)
52.             {
53.                 //Заполнение столбцов
54.                 for (int i = 0; i < comp.Properties.Count; i++)
55.                     dat.Columns.Add( new DataColumn() { ColumnName =
comp.Properties[i].First.First, Caption = comp.Properties[i].First.Second });
56.
57.                 //Заполнение строк
58.                 for (int i = 0; i < comp.Properties.Count; i++)
59.                     for (int j = 0; j < dat.Columns.Count; j++)
60.                         if (dat.Columns[j].ColumnName ==
comp.Properties[i].First.First)
61.                             for (int k = 0; k <
comp.Properties[i].Second.Count; k++)
62.                             {
63.                                 if (dat.Rows.Count <= k)
64.                                     dat.Rows.Add();
65.                                 dat.Rows[k][j] =
comp.Properties[i].Second[k];
66.                             }
67.
68.                 //Визуализация столбцов
69.                 foreach (DataColumn i in dat.Columns)
70.                 {
71.                     bool r = (i.ColumnName[0] == '=') ? true : false;
72.                     ComposSystemTable.Columns.Add(new DataGridTextColumn()
73.                     {
74.                         Header = (i.Caption == "" || i.Caption == " ") ?
i.ColumnName : (i.ColumnName[0] == '=') ? i.Caption + i.ColumnName :
i.ColumnName + ", " + i.Caption,
75.                         Binding = new Binding "[" +
ComposSystemTable.Columns.Count + "]" ),
76.                         IsReadOnly = r

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

77.         });
78.     }
79. }
80. }
81. else
82. {
83.     this.Title = "Таблица системы " + elem;
84.
85.     dat = MendeleevTable.BinarySystem.Find(x => x.TableName ==
elem);
86.
87.     if (dat == null)
88.     {
89.         dat = new DataTable() { TableName = elem };
90.         dat.Columns.Add(new DataColumn("X") { Caption = "x" });
91.         ComposSystemTable.Columns.Add(new DataGridTextColumn()
92.         {
93.             Header = "X",
94.             Binding = new Binding("[0]"),
95.             IsReadOnly = false,
96.             Width = 100
97.         });
98.     }
99.     else
100.    {
101.        foreach (DataColumn i in dat.Columns)
102.        {
103.            bool r = (i.ColumnName[0] == '=') ? true : false;
104.            ComposSystemTable.Columns.Add(new
DataGridTextColumn()
105.            {
106.                Header = (i.Caption == "" || i.Caption == "
") ? i.ColumnName : (i.ColumnName[0] == '=') ? i.Caption + i.ColumnName :
i.ColumnName + ", " + i.Caption,
107.                Binding = new Binding("[ " +
ComposSystemTable.Columns.Count + "]" ),
108.                IsReadOnly = r
109.            });
110.        }
111.    }
112. }
113. ComposSystemTable.ItemsSource = dat.DefaultView;

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```
114.
115.             if (f && dat.Columns.Count > 0 || !f && dat.Columns.Count >
116.                 1)
117.                 DelColumn.IsEnabled = true;
118.             if (dat.Rows.Count > 0)
119.                 DelSelectedRows.IsEnabled = true;
120.         }
121.         /// <summary>
122.         /// Добавляет текстовый столбец в таблицу
123.         /// </summary>
124.         private void AddColumn_Click(object sender, RoutedEventArgs e)
125.         {
126.             DelColumn.IsEnabled = DataGridHelper.AddColumn(ref
127.                 ComposSystemTable, ref dat, f);
128.         }
129.         /// <summary>
130.         /// Добавляет строку в таблицу
131.         /// </summary>
132.         private void AddRow_Click(object sender, RoutedEventArgs e)
133.         {
134.             dat.Rows.Add();
135.             if (dat.Rows.Count > 0)
136.                 DelSelectedRows.IsEnabled = true;
137.             if (!f)
138.                 dat.Rows[dat.Rows.Count - 1][0] = 0;
139.             CollectionViewSource.GetDefaultView(ComposSystemTable.ItemsSource).Refresh();
140.         }
141.         /// <summary>
142.         /// Удаляет столбец в таблице
143.         /// </summary>
144.         private void DelColumn_Click(object sender, RoutedEventArgs e)
145.         {
146.             ComposSystemTable.Columns.RemoveAt(ComposSystemTable.Columns.Count - 1);
147.             if (f && dat.Columns.Count <= 1 || !f && dat.Columns.Count <=
148.                 2)
149.                 DelColumn.IsEnabled = false;
150.             dat.Columns.RemoveAt(dat.Columns.Count - 1);
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

151.         }
152.
153.         /// <summary>
154.         /// Удаляет выделенные строки в таблице
155.         /// </summary>
156.         private void DelSelectedRows_Click(object sender, RoutedEventArgs
            e)
157.         {
158.             try
159.             {
160.                 while (ComposSystemTable.SelectedItems.Count > 0)
161.                 {
162.                     int selectedIndex = ComposSystemTable.SelectedIndex;
163.
164.                     DrawingClasses.CollapseGraph.RemoveSelectedPoint(selectedIndex);
165.                     dat.Rows.RemoveAt(selectedIndex);
166.                     ComposSystemTable.Items.Refresh();
167.                 }
168.             }
169.             catch (Exception)
170.             {
171.                 MessageBox.Show("Невозможно удалить этот элемент!",
172. "Error", MessageBoxButton.OK, MessageBoxImage.Error);
173.             }
174.             if (dat.Rows.Count == 0)
175.                 DelSelectedRows.IsEnabled = false;
176.         }
177.
178.         /// <summary>
179.         /// Запускает расчет формул и сохраняет данные в файл
180.         /// </summary>
181.         private void Calculate_Click(object sender, RoutedEventArgs e)
182.         {
183.             ComposSystemTable.IsReadOnly = true;
184.             EditTable.Visibility = Visibility.Visible;
185.             AddColumn.Visibility = Visibility.Hidden;
186.             AddRow.Visibility = Visibility.Hidden;
187.             DelColumn.Visibility = Visibility.Hidden;
188.             DelSelectedRows.Visibility = Visibility.Hidden;
189.             Calculate.Visibility = Visibility.Hidden;
190.             AddFormul.Visibility = Visibility.Hidden;
191.             if (!f)

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

32
RU.17701729.04.15-01 12 01-1

```
190.             DomeOfDecayWindowOpen.Visibility = Visibility.Visible;
191.             MathParser.f = f;
192.
193.             try
194.             {
195.                 //Расчет формул
196.                 for (int i = f?0:1; i < dat.Columns.Count; i++)
197.                     for (int u = 0; u < dat.Rows.Count; u++)
198.                         if (dat.Columns[i].ColumnName[0] == '=')
199.                             dat.Rows[u][i] = String.Format("{0:f4}",
200.                                 MathParser.Parse(dat.Columns[i].ColumnName.Substring(1), ref dat, u));
201.             }
202.             catch (Exception ex)
203.             {
204.                 if (ex.InnerException.Message == "MyException")
205.                     MessageBox.Show(ex.Message, "Error",
206.                         MessageBoxButton.OK, MessageBoxImage.Error);
207.                 else
208.                     MessageBox.Show("Неверный формат формулы!", "Error",
209.                         MessageBoxButton.OK, MessageBoxImage.Error);
210.             }
211.             dat.AcceptChanges();
212.
213.             if (f)
214.             {
215.                 List<Pair<Pair<string, string>, List<string>>> prop = new
216.                     List<Pair<Pair<string, string>, List<string>>>();
217.
218.                 if (prop == null)
219.                     prop = new List<Pair<Pair<string, string>,
220.                         List<string>>>();
221.
222.                 for (int i = 0; i < dat.Columns.Count; i++)
223.                 {
224.                     prop.Add(new Pair<Pair<string, string>,
225.                         List<string>>(new Pair<string, string>(dat.Columns[i].ColumnName,
226.                             dat.Columns[i].Caption), new List<string>()));
227.                     for (int j = 0; j < dat.Rows.Count; j++)
228.                         prop[i].Second.Add(dat.Rows[j][i].ToString());
229.                 }
230.             }
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

224.                Composition el = MendeleevTable.Compos.Find(x => x.Name
                == elem);
225.                if (el != null)
226.                    MendeleevTable.Compos.Find(x => x.Name ==
                elem).Properties = new List<Pair<Pair<string, string>, List<string>>>();
227.                else
228.                    MendeleevTable.Compos.Add(new Composition() { Name =
                elem });
229.
230.                MendeleevTable.Compos.Find(x => x.Name ==
                elem).Properties = prop;
231.
232.                DataGridHelper.Serialize("Compositions.xml", ref
                MendeleevTable.Compos);
233.            }
234.            else
235.            {
236.
                MendeleevTable.BinarySystem.Remove(MendeleevTable.BinarySystem.Find(x =>
                x.TableName == elem));
237.                MendeleevTable.BinarySystem.Add(dat);
238.
239.                DataGridHelper.Serialize("BinarySystems.xml", ref
                MendeleevTable.BinarySystem);
240.            }
241.        }
242.
243.        /// <summary>
244.        /// Добавляет столбец-формулу в таблицу
245.        /// </summary>
246.        private void AddFormul_Click(object sender, RoutedEventArgs e)
247.        {
248.            FormulaInput form = new FormulaInput();
249.            form.ShowDialog();
250.            try
251.            {
252.                if (form.formula != "")
253.                {
254.                    if (form.symbol != "" && form.symbol != " ")
255.                        foreach (DataColumn v in dat.Columns)
256.                            if (v.Caption == form.symbol)

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```
257.                                     throw new DuplicateNameException("Такая
    формула уже принадлежит данной таблице!", new Exception("MyException"));
258.
259.                                     DataColumn col = new DataColumn('=' + form.formula) {
    Caption = form.symbol };
260.                                     dat.Columns.Add(col);
261.                                     ComposSystemTable.Columns.Add(new
    DataGridViewTextBoxColumn()
262.                                     {
263.                                         Header = form.symbol + '=' + form.formula,
264.                                         Binding = new Binding "[" +
    ComposSystemTable.Columns.Count + "]",
265.                                         IsReadOnly = true
266.                                     });
267.                                     for (int u = 0; u < dat.Rows.Count; u++)
268.                                         dat.Rows[u][dat.Columns.Count - 1] =
    MathParser.Parse(dat.Columns[dat.Columns.Count - 1].ColumnName.Substring(1),
    ref dat, u);
269.                                     }
270.                                     }
271.                                     catch (Exception ex)
272.                                     {
273.                                         if (ex.InnerException != null &&
    ex.InnerException.Message == "MyException")
274.                                             MessageBox.Show(ex.Message, "Error",
    MessageBoxButtons.OK, MessageBoxIcon.Error);
275.                                         else
276.                                             MessageBox.Show("Неверный формат формулы или она уже
    содержится в данной таблице!", "Error", MessageBoxButtons.OK,
    MessageBoxIcon.Error);
277.                                     }
278.                                     if (f && dat.Columns.Count > 0 || !f && dat.Columns.Count >
    1)
279.                                         DelColumn.IsEnabled = true;
280.                                     }
281.
282.                                     private void Window_Closing(object sender,
    System.ComponentModel.CancelEventArgs e)
283.                                     {
284.                                         if (EditTable.Visibility == Visibility.Hidden)
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

35
RU.17701729.04.15-01 12 01-1

```
285.             if (MessageBox.Show("Вы точно хотите закрыть окно? Все
                несохраненные данные будут удалены!", "", MessageBoxButton.YesNo,
                MessageBoxImage.Question) == MessageBoxResult.No)
286.                 e.Cancel = true;
287.             else
288.                 dat.RejectChanges();
289.         }
290.
291.         private void Window_SizeChanged(object sender,
                SizeChangedEventArgs e)
292.         {
293.             if (e.PreviousSize.Width != 0)
294.             {
295.                 AddRow.Width += (e.NewSize.Width - e.PreviousSize.Width)
                / 6;
296.                 AddColumn.Width += (e.NewSize.Width -
                e.PreviousSize.Width) / 6;
297.                 DelColumn.Width += (e.NewSize.Width -
                e.PreviousSize.Width) / 6;
298.                 DelSelectedRows.Width += (e.NewSize.Width -
                e.PreviousSize.Width) / 6;
299.                 AddFormul.Width += (e.NewSize.Width -
                e.PreviousSize.Width) / 6;
300.                 Calculate.Width += (e.NewSize.Width -
                e.PreviousSize.Width) / 6;
301.                 AddRow.RenderTransform = new TranslateTransform(360 +
                (e.NewSize.Width - 1050) / 3, 0);
302.                 AddColumn.RenderTransform = new TranslateTransform(190 +
                (e.NewSize.Width - 1050) / 6, 0);
303.                 DelSelectedRows.RenderTransform = new
                TranslateTransform(700 + (e.NewSize.Width - 1050) / 1.5, 0);
304.                 DelColumn.RenderTransform = new TranslateTransform(530 +
                (e.NewSize.Width - 1050) / 2, 0);
305.                 Calculate.RenderTransform = new TranslateTransform(870 +
                (e.NewSize.Width - 1050) / 1.2, 0);
306.             }
307.         }
308.
309.         private void ComposSystemTable_CellEditEnding(object sender,
                DataGridCellEditEndingEventArgs e)
310.         {
311.             string str1 = (e.EditingElement as TextBox).Text;
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

312.         double i = 0;
313.         str1 = str1.Replace('.', ',');
314.         if (!double.TryParse(str1, out i))
315.             str1 = StringHelper.DoString(str1);
316.         else
317.             str1 = String.Format("{0:f4}", i);
318.
319.         if (e.Column.DisplayIndex == 0)
320.             str1 = i.ToString();
321.
322.         dat.Rows[e.Row.GetIndex()][e.Column.DisplayIndex] = str1;
323.         (e.EditingElement as TextBox).Text = str1;
324.     }
325.
326.     /// <summary>
327.     /// Позволяет редактировать таблицу
328.     /// </summary>
329.     private void EditTable_Click(object sender, RoutedEventArgs e)
330.     {
331.         CompoSystemTable.IsReadOnly = false;
332.         EditTable.Visibility = Visibility.Hidden;
333.         AddColumn.Visibility = Visibility.Visible;
334.         AddRow.Visibility = Visibility.Visible;
335.         DelColumn.Visibility = Visibility.Visible;
336.         DelSelectedRows.Visibility = Visibility.Visible;
337.         Calculate.Visibility = Visibility.Visible;
338.         AddFormul.Visibility = Visibility.Visible;
339.         if (!f)
340.             DomeOfDecayWindowOpen.Visibility = Visibility.Hidden;
341.     }
342.
343.     /// <summary>
344.     /// Открывает окно построения купола распада
345.     /// </summary>
346.     private void DomeOfDecayWindowOpen_Click(object sender,
        RoutedEventArgs e)
347.     {
348.         new DomeOfDecay(elem).ShowDialog();
349.     }
350. }
351. }
352.

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

4. Класс CreateComposSystem:

- CreateComposSystem.xaml**

```

1. <Window x:Class="Activision_Mendelejev_table.CreateComposSystem"
2.     xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
3.     xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
4.     xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
5.     xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
6.     mc:Ignorable="d"
7.     Icon="icon.jpeg"
8.     Title="Выберите соединение" Height="300" Width="300"
   ResizeMode="NoResize" WindowStartupLocation="CenterScreen">
9.     <Grid>
10.         <Button Name="Complete" Content="Готово" Click="Complete_Click"
   Width="150" Height="40" RenderTransformOrigin="0.5,0.5"
   HorizontalAlignment="Center" VerticalAlignment="Bottom" FontSize="16" >
11.             <Button.RenderTransform>
12.                 <TransformGroup>
13.                     <ScaleTransform/>
14.                     <SkewTransform/>
15.                     <RotateTransform/>
16.                     <TranslateTransform Y="-15"/>
17.                 </TransformGroup>
18.             </Button.RenderTransform>
19.         </Button>
20.         <TextBox Name="NewComposSystem" Width="250" Height="30"
   RenderTransformOrigin="0.5,0.5" FontSize="20">
21.             <TextBox.RenderTransform>
22.                 <TransformGroup>
23.                     <ScaleTransform/>
24.                     <SkewTransform/>
25.                     <RotateTransform/>
26.                     <TranslateTransform Y="0"/>
27.                 </TransformGroup>
28.             </TextBox.RenderTransform>
29.         </TextBox>
30.         <Label Name="label" Height="30" RenderTransformOrigin="0.5,0.5"
   HorizontalAlignment="Center">
31.             <Label.RenderTransform>
32.                 <TransformGroup>
33.                     <ScaleTransform/>
34.                     <SkewTransform/>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

35.             <RotateTransform/>
36.             <TranslateTransform Y="-30"/>
37.         </TransformGroup>
38.     </Label.RenderTransform>
39. </Label>
40.     <ComboBox Name="ComposSystem" Width="250" Height="40" FontSize="20"
        VerticalAlignment="Top" RenderTransformOrigin="0.5,0.5"
        SelectionChanged="Compos_SelectionChanged" >
41.         <ComboBox.RenderTransform>
42.             <TransformGroup>
43.                 <ScaleTransform/>
44.                 <SkewTransform/>
45.                 <RotateTransform/>
46.                 <TranslateTransform Y="15"/>
47.             </TransformGroup>
48.         </ComboBox.RenderTransform>
49.     </ComboBox>
50. </Grid>
51.</Window>

```

- **CreateComposSystem.cs**

```

1. using System.Windows;
2. using Activision_Mendeleyev_table.HelperClasses;
3.
4. namespace Activision_Mendeleyev_table
5. {
6.     /// <summary>
7.     /// Логика взаимодействия для CreateComposSystem.xaml
8.     /// </summary>
9.     public partial class CreateComposSystem : Window
10.    {
11.        /// <summary>
12.        /// Флаг: true - соединение, false - система
13.        /// </summary>
14.        private bool f;
15.
16.        /// <summary>
17.        /// Конструктор, инициализирующий окно создания/выбора
        соединения(системы соединений)
18.        /// </summary>
19.        /// <param name="f">флаг: true - соединение, false - система</param>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

20.         public CreateComposSystem(bool f)
21.         {
22.             InitializeComponent();
23.
24.             this.f = f;
25.
26.             if (f)
27.             {
28.                 this.Title = "Выберите/введите соединение";
29.                 label.Content = "Введите новое соединения";
30.                 for (int i = 0; i < MendeleevTable.Compos.Count; i++)
31.                     ComposSystem.Items.Add(MendeleevTable.Compos[i].Name);
32.             }
33.             else
34.             {
35.                 label.Content = "Введите новую систему соединений";
36.                 this.Title = "Выберите/введите систему соединений";
37.                 for (int i = 0; i < MendeleevTable.BinarySystem.Count; i++)
38.                     ComposSystem.Items.Add(MendeleevTable.BinarySystem[i].TableName);
39.             }
40.
41.             if (ComposSystem.Items.Count > 0)
42.                 ComposSystem.SelectedIndex = 0;
43.         }
44.
45.         /// <summary>
46.         /// Открывает таблицу созданного/выбранного соединения (системы
соединений)
47.         /// </summary>
48.         private void Complete_Click(object sender, RoutedEventArgs e)
49.         {
50.             this.Hide();
51.             new
ComposSystemContent(StringHelper.DoString(NewComposSystem.Text),
f).ShowDialog();
52.             this.Close();
53.         }
54.
55.         /// <summary>
56.         /// Дублирует соединение (систему соединений) из ComboBox в TextBox
57.         /// </summary>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата


```

58.         private void Compos_SelectionChanged(object sender,
           System.Windows.Controls.SelectionChangedEventArgs e)
59.         {
60.             NewComposSystem.Text = e.AddedItems[0] as string;
61.         }
62.     }
63. }
64.

```

5. Класс ColumnAddWindow:

- ColumnAddWindow.xaml

```

1. <Window x:Class="Activision_Mendelejev_table.ColumnAddWindow"
2.         xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
3.         xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
4.         xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
5.         xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
6.         mc:Ignorable="d"
7.         Icon="icon.jpeg"
8.         Title="Введите название столбца " Height="150" Width="265"
           ResizeMode="NoResize" WindowStartupLocation="CenterScreen">
9.     <Grid>
10.         <Button Name="Create" Content="Создать" HorizontalAlignment="Left"
           Margin="20,70,0,0" VerticalAlignment="Top" Width="105" Click="Create_Click"
           Height="38"/>
11.         <TextBox Name="ColumnName" HorizontalAlignment="Left" Height="20"
           Margin="20,10,0,0" TextWrapping="Wrap" Text="New Column"
           VerticalAlignment="Top" Width="215"/>
12.         <TextBox Name="ColumnSymbol" HorizontalAlignment="Left" Height="20"
           Margin="120,40,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="115"
           ToolTip="Если обозначение не требуется, оставьте поле пустым"/>
13.         <Label HorizontalAlignment="Left" Height="30" Margin="20,35,0,0"
           Content="Обозначение: " VerticalAlignment="Top"/>
14.         <Button Name="Cancel" Content="Отмена" HorizontalAlignment="Left"
           Margin="130,70,0,0" VerticalAlignment="Top" Width="105" Click="Cancel_Click"
           Height="38"/>
15.     </Grid>
16. </Window>
17.

```

- ColumnAddWindow.cs

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```
1. using System.Windows;
2.
3. namespace Activision_Mendelejev_table
4. {
5.     /// <summary>
6.     /// Логика взаимодействия для ColumnAddWindow.xaml
7.     /// </summary>
8.     public partial class ColumnAddWindow : Window
9.     {
10.         /// <summary>
11.         /// Название столбца
12.         /// </summary>
13.         public string name { get; private set; }
14.
15.         /// <summary>
16.         /// Обозначение свойства, значения которого будут находиться в данном
17.         /// столбце
18.         public string symbol { get; private set; }
19.
20.         /// <summary>
21.         /// Конструктор, инициализирующий окно создания текстового столбца
22.         /// </summary>
23.         public ColumnAddWindow()
24.         {
25.             InitializeComponent();
26.             name = "";
27.             symbol = "";
28.         }
29.
30.         /// <summary>
31.         /// Создание текстового столбца
32.         /// </summary>
33.         private void Create_Click(object sender, RoutedEventArgs e)
34.         {
35.             name = ColumnName.Text;
36.             symbol = ColumnSymbol.Text;
37.             Close();
38.         }
39.
40.         /// <summary>
41.         /// Отмена изменений и закрытие окна
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

42.      /// </summary>
43.      private void Cancel_Click(object sender, RoutedEventArgs e)
44.      {
45.          Close();
46.      }
47.  }
48. }
49.

```

6. Класс FormulaInput:

- FormulaInput.xaml

```

1. <Window x:Class="Activision_Mendelejev_table.FormulaInput"
2.      xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
3.      xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
4.      xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
5.      xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
6.      mc:Ignorable="d"
7.      Icon="icon.jpeg"
8.      Title="Введите формулу" Height="150" Width="370" ResizeMode="NoResize"
   WindowStartupLocation="CenterScreen">
9.     <Grid>
10.        <Button Name="Create" Content="Создать" Width="100"
   Click="Create_Click" Height="38" VerticalAlignment="Bottom"
   HorizontalAlignment="Left" RenderTransformOrigin="0.5,0.5">
11.            <Button.RenderTransform>
12.                <TransformGroup>
13.                    <ScaleTransform/>
14.                    <SkewTransform/>
15.                    <RotateTransform/>
16.                    <TranslateTransform X="15" Y="-15"/>
17.                </TransformGroup>
18.            </Button.RenderTransform>
19.        </Button>
20.        <Button Name="Cancel" Content="Отмена" Width="100"
   Click="Cancel_Click" Height="38" VerticalAlignment="Bottom"
   HorizontalAlignment="Right" RenderTransformOrigin="0.5,0.5">
21.            <Button.RenderTransform>
22.                <TransformGroup>
23.                    <ScaleTransform/>
24.                    <SkewTransform/>
25.                    <RotateTransform/>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

26.             <TranslateTransform X="-15" Y="-15"/>
27.         </TransformGroup>
28.     </Button.RenderTransform>
29. </Button>
30.     <Button Name="OpenFormuls" Content="Список формул" Width="100"
Click="OpenFormuls_Click" Height="38" VerticalAlignment="Bottom"
HorizontalAlignment="Center" RenderTransformOrigin="0.5,0.5">
31.         <Button.RenderTransform>
32.             <TransformGroup>
33.                 <ScaleTransform/>
34.                 <SkewTransform/>
35.                 <RotateTransform/>
36.                 <TranslateTransform Y="-15"/>
37.             </TransformGroup>
38.         </Button.RenderTransform>
39. </Button>
40.     <TextBox Name="NewFormula" Width="270" Height="30"
RenderTransformOrigin="0.5,0.5" FontSize="20" VerticalAlignment="Top">
41.         <TextBox.RenderTransform>
42.             <TransformGroup>
43.                 <ScaleTransform/>
44.                 <SkewTransform/>
45.                 <RotateTransform/>
46.                 <TranslateTransform X="30" Y="20"/>
47.             </TransformGroup>
48.         </TextBox.RenderTransform>
49. </TextBox>
50.     <Label Content="=" Height="30" RenderTransformOrigin="0.5,0.5"
HorizontalAlignment="Left" VerticalAlignment="Top">
51.         <Label.RenderTransform>
52.             <TransformGroup>
53.                 <ScaleTransform/>
54.                 <SkewTransform/>
55.                 <RotateTransform/>
56.                 <TranslateTransform X="55" Y="20"/>
57.             </TransformGroup>
58.         </Label.RenderTransform>
59. </Label>
60.     <TextBox Name="FormulaSymbol" Width="40" Height="30"
RenderTransformOrigin="0.5,0.5" FontSize="20" HorizontalAlignment="Left"
VerticalAlignment="Top" >
61.         <TextBox.RenderTransform>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

62.         <TransformGroup>
63.             <ScaleTransform/>
64.             <SkewTransform/>
65.             <RotateTransform/>
66.             <TranslateTransform X="15" Y="20"/>
67.         </TransformGroup>
68.     </TextBox.RenderTransform>
69. </TextBox>
70. </Grid>
71.</Window>

```

- **FormulaInput.cs**

```

1. using System.Windows;
2. using Activision_Mendeleyev_table.HelperClasses;
3.
4. namespace Activision_Mendeleyev_table
5. {
6.     /// <summary>
7.     /// Логика взаимодействия для FormulaInput.xaml
8.     /// </summary>
9.     public partial class FormulaInput : Window
10.    {
11.        /// <summary>
12.        /// Строка-формула
13.        /// </summary>
14.        public string formula { get; private set; }
15.
16.        /// <summary>
17.        /// Обозначение формулы
18.        /// </summary>
19.        public string symbol { get; private set; }
20.
21.        /// <summary>
22.        /// Конструктор, инициализирующий окно создания столбца-формулы
23.        /// </summary>
24.        public FormulaInput()
25.        {
26.            InitializeComponent();
27.            formula = "";
28.            symbol = "";
29.        }

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

30.
31.     /// <summary>
32.     /// Создание столбца-формулы
33.     /// </summary>
34.     private void Create_Click(object sender, RoutedEventArgs e)
35.     {
36.         formula = StringHelper.DoString(NewFormula.Text);
37.         symbol = StringHelper.DoString(FormulaSymbol.Text);
38.         Close();
39.     }
40.
41.     /// <summary>
42.     /// Отмена изменений и закрытие окна
43.     /// </summary>
44.     private void Cancel_Click(object sender, RoutedEventArgs e)
45.     {
46.         Close();
47.     }
48.
49.     /// <summary>
50.     /// Просмотр списка формул
51.     /// </summary>
52.     private void OpenFormuls_Click(object sender, RoutedEventArgs e)
53.     {
54.         new FormulaList().Show();
55.     }
56. }
57. }
58.

```

7. Класс FormulaList:

- FormulaList.xaml

```

1. <Window x:Class="Activision_Mendelejev_table.FormulaList"
2.     xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
3.     xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
4.     xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
5.     xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
6.     mc:Ignorable="d"
7.     Icon="icon.jpeg"
8.     Title="Формулы и столбцы" ResizeMode="NoResize" Height="450"
    Width="800" WindowStartupLocation="CenterScreen">

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

9.      <Grid>
10.          <DataGrid x:Name="formuls" CanUserSortColumns="False"
              CanUserReorderColumns="False" CanUserResizeColumns="False" ColumnWidth="*"
11.              AutoGenerateColumns="False" CanUserAddRows="False"
              CanUserDeleteRows="False" CanUserResizeRows="False"/>
12.      </Grid>
13.</Window>

```

• FormulaList.cs

```

1. using System.Data;
2. using System.Windows;
3.
4. namespace Activision_Mendeleev_table
5. {
6.     /// <summary>
7.     /// Логика взаимодействия для FormulaList.xaml
8.     /// </summary>
9.     public partial class FormulaList : Window
10.    {
11.        public FormulaList()
12.        {
13.            InitializeComponent();
14.            DataTable dat = new DataTable();
15.            dat.Columns.Add("Обозначение");
16.            dat.Columns.Add("Название/формула");
17.            foreach (DataColumn i in dat.Columns)
18.            {
19.                formuls.Columns.Add(new
                System.Windows.Controls.DataGridTextColumn()
20.                {
21.                    Header = i.ColumnName,
22.                    Binding = new System.Windows.Data.Binding "[" +
                formuls.Columns.Count + "]"
23.                });
24.            }
25.            for (int i = 0; i <
                MendeleevTable.Elems[0].DataTable.Columns.Count; i++)
26.            {
27.                dat.Rows.Add(MendeleevTable.Elems[0].DataTable.Columns[i].Caption != null?
                MendeleevTable.Elems[0].DataTable.Columns[i].Caption:"",
                MendeleevTable.Elems[0].DataTable.Columns[i].ColumnName);
28.            }
29.        }
30.    }
31. }

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

27.         for (int i = 0; i < MendeleevTable.Compos.Count; i++)
28.             for (int j = 0; j < MendeleevTable.Compos[i].Properties.Count;
                j++)
29.
    dat.Rows.Add(MendeleevTable.Compos[i].Properties[j].First.First != null ?
MendeleevTable.Compos[i].Properties[j].First.Second : "",
    (MendeleevTable.Compos[i].Properties[j].First.First[0] !=
    '=')?MendeleevTable.Compos[i].Properties[j].First.First:
MendeleevTable.Compos[i].Properties[j].First.First.Substring(1));
30.         for (int i = 0; i < MendeleevTable.BinarySystem.Count; i++)
31.             for (int j = 0; j <
MendeleevTable.BinarySystem[i].Columns.Count; j++)
32.
    dat.Rows.Add(MendeleevTable.BinarySystem[i].Columns[j].Caption != null ?
MendeleevTable.BinarySystem[i].Columns[j].Caption : "",
    (MendeleevTable.BinarySystem[i].Columns[j].ColumnName[0] !=
    '=')?MendeleevTable.BinarySystem[i].Columns[j].ColumnName:
MendeleevTable.BinarySystem[i].Columns[j].ColumnName.Substring(1));
33.
34.         formuls.ItemsSource = dat.DefaultView;
35.     }
36. }
37. }
38.

```

8. Класс DataSettings:

- DataSettings.xaml

```

1. <Window x:Class="Activision_Mendeleev_table.DataSettings"
2.     xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
3.     xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
4.     xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
5.     xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
6.     mc:Ignorable="d"
7.     Icon="icon.jpeg"
8.     Title="Настройка данных" WindowStartupLocation="CenterScreen"
    Height="600" Width="600" Closing="Window_Closing" ResizeMode="NoResize" >
9.     <Grid>
10.         <TextBlock Text="Первый химический элемент:" TextWrapping="Wrap"
            RenderTransformOrigin="0.5,0.5" FontSize="20" VerticalAlignment="Top"
            HorizontalAlignment="Left" Width="300" Height="40" >
11.             <TextBlock.RenderTransform>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

48
RU.17701729.04.15-01 12 01-1

```

12.      <TransformGroup>
13.      <ScaleTransform/>
14.      <SkewTransform/>
15.      <RotateTransform/>
16.      <TranslateTransform X="20" Y="70"/>
17.      </TransformGroup>
18.      </TextBlock.RenderTransform>
19.      </TextBlock>
20.      <TextBlock Text="Символ химиче-ского элемента" TextWrapping="Wrap"
RenderTransformOrigin="0.5,0.5" FontSize="20" VerticalAlignment="Top"
HorizontalAlignment="Left" Width="160" Height="55" >
21.      <TextBlock.RenderTransform>
22.      <TransformGroup>
23.      <ScaleTransform/>
24.      <SkewTransform/>
25.      <RotateTransform/>
26.      <TranslateTransform X="310" Y="10"/>
27.      </TransformGroup>
28.      </TextBlock.RenderTransform>
29.      </TextBlock>
30.      <TextBlock Text="Номер строки" TextWrapping="Wrap"
RenderTransformOrigin="0.5,0.5" FontSize="20" VerticalAlignment="Top"
HorizontalAlignment="Left" Width="100" Height="55" >
31.      <TextBlock.RenderTransform>
32.      <TransformGroup>
33.      <ScaleTransform/>
34.      <SkewTransform/>
35.      <RotateTransform/>
36.      <TranslateTransform X="490" Y="10"/>
37.      </TransformGroup>
38.      </TextBlock.RenderTransform>
39.      </TextBlock>
40.      <TextBox x:Name="elemA" Text="A" FontSize="20"
RenderTransformOrigin="0.5,0.5" Width="100" Height="30"
HorizontalAlignment="Left" VerticalAlignment="Top">
41.      <TextBox.RenderTransform>
42.      <TransformGroup>
43.      <ScaleTransform/>
44.      <SkewTransform/>
45.      <RotateTransform/>
46.      <TranslateTransform X="340" Y="70"/>
47.      </TransformGroup>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

48.         </TextBox.RenderTransform>
49.     </TextBox>
50.     <TextBox x:Name="numA" Text="0" FontSize="20"
RenderTransformOrigin="0.5,0.5" Width="100" Height="30"
HorizontalAlignment="Left" VerticalAlignment="Top">
51.         <TextBox.RenderTransform>
52.             <TransformGroup>
53.                 <ScaleTransform/>
54.                 <SkewTransform/>
55.                 <RotateTransform/>
56.                 <TranslateTransform X="470" Y="70"/>
57.             </TransformGroup>
58.         </TextBox.RenderTransform>
59.     </TextBox>
60.     <TextBlock Text="Второй химический элемент:" TextWrapping="Wrap"
FontSize="20" VerticalAlignment="Top" HorizontalAlignment="Left"
RenderTransformOrigin="0.5,0.5" Width="300" Height="40">
61.         <TextBlock.RenderTransform>
62.             <TransformGroup>
63.                 <ScaleTransform/>
64.                 <SkewTransform/>
65.                 <RotateTransform/>
66.                 <TranslateTransform X="20" Y="110"/>
67.             </TransformGroup>
68.         </TextBlock.RenderTransform>
69.     </TextBlock>
70.     <TextBox x:Name="elemB" Text="B" FontSize="20"
RenderTransformOrigin="0.5,0.5" VerticalAlignment="Top"
HorizontalAlignment="Left" Height="30" Width="100">
71.         <TextBox.RenderTransform>
72.             <TransformGroup>
73.                 <ScaleTransform/>
74.                 <SkewTransform/>
75.                 <RotateTransform/>
76.                 <TranslateTransform X="340" Y="110"/>
77.             </TransformGroup>
78.         </TextBox.RenderTransform>
79.     </TextBox>
80.     <TextBox x:Name="numB" Text="0" FontSize="20"
RenderTransformOrigin="0.5,0.5" VerticalAlignment="Top"
HorizontalAlignment="Left" Height="30" Width="100">
81.         <TextBox.RenderTransform>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

50
RU.17701729.04.15-01 12 01-1

```

82.      <TransformGroup>
83.      <ScaleTransform/>
84.      <SkewTransform/>
85.      <RotateTransform/>
86.      <TranslateTransform X="470" Y="110"/>
87.      </TransformGroup>
88.      </TextBox.RenderTransform>
89.      </TextBox>
90.      <TextBlock Text="Общий химический элемент:" TextWrapping="Wrap"
      FontSize="20" VerticalAlignment="Top" HorizontalAlignment="Left"
      RenderTransformOrigin="0.5,0.5" Width="300" Height="40">
91.      <TextBlock.RenderTransform>
92.      <TransformGroup>
93.      <ScaleTransform/>
94.      <SkewTransform/>
95.      <RotateTransform/>
96.      <TranslateTransform X="20" Y="150"/>
97.      </TransformGroup>
98.      </TextBlock.RenderTransform>
99.      </TextBlock>
100.     <TextBox x:Name="elemX" Text="X" FontSize="20"
      RenderTransformOrigin="0.5,0.5" VerticalAlignment="Top"
      HorizontalAlignment="Left" Height="30" Width="100">
101.     <TextBox.RenderTransform>
102.     <TransformGroup>
103.     <ScaleTransform/>
104.     <SkewTransform/>
105.     <RotateTransform/>
106.     <TranslateTransform X="340" Y="150"/>
107.     </TransformGroup>
108.     </TextBox.RenderTransform>
109.     </TextBox>
110.     <TextBox x:Name="numX" Text="0" FontSize="20"
      RenderTransformOrigin="0.5,0.5" VerticalAlignment="Top"
      HorizontalAlignment="Left" Height="30" Width="100">
111.     <TextBox.RenderTransform>
112.     <TransformGroup>
113.     <ScaleTransform/>
114.     <SkewTransform/>
115.     <RotateTransform/>
116.     <TranslateTransform X="470" Y="150"/>
117.     </TransformGroup>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

51
RU.17701729.04.15-01 12 01-1

```

118.         </TextBox.RenderTransform>
119.     </TextBox>
120.     <TextBlock Text="Обозначение радиуса иона элемента в таблице
данных:" TextWrapping="Wrap" FontSize="20" VerticalAlignment="Top"
HorizontalAlignment="Left" RenderTransformOrigin="0.5,0.5" Width="347"
Height="55">
121.         <TextBlock.RenderTransform>
122.             <TransformGroup>
123.                 <ScaleTransform/>
124.                 <SkewTransform/>
125.                 <RotateTransform/>
126.                 <TranslateTransform X="20" Y="190"/>
127.             </TransformGroup>
128.         </TextBlock.RenderTransform>
129.     </TextBlock>
130.     <TextBox x:Name="r" Text="R(i)" FontSize="20"
RenderTransformOrigin="0.5,0.5" VerticalAlignment="Top"
HorizontalAlignment="Left" Height="30" Width="100">
131.         <TextBox.RenderTransform>
132.             <TransformGroup>
133.                 <ScaleTransform/>
134.                 <SkewTransform/>
135.                 <RotateTransform/>
136.                 <TranslateTransform X="400" Y="205"/>
137.             </TransformGroup>
138.         </TextBox.RenderTransform>
139.     </TextBox>
140.     <TextBlock Text="Обозначение электроотрицательности элемента в
таблице данных:" TextWrapping="Wrap" FontSize="20" VerticalAlignment="Top"
HorizontalAlignment="Left" RenderTransformOrigin="0.5,0.5" Width="365"
Height="55">
141.         <TextBlock.RenderTransform>
142.             <TransformGroup>
143.                 <ScaleTransform/>
144.                 <SkewTransform/>
145.                 <RotateTransform/>
146.                 <TranslateTransform X="20" Y="260"/>
147.             </TransformGroup>
148.         </TextBlock.RenderTransform>
149.     </TextBlock>
150.     <TextBlock Text="Обозначение формального заряда элемента в
таблице данных:" TextWrapping="Wrap" FontSize="20" VerticalAlignment="Top"

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

HorizontalAlignment="Left" RenderTransformOrigin="0.5,0.5" Width="365"
Height="55">
151.         <TextBlock.RenderTransform>
152.         <TransformGroup>
153.         <ScaleTransform/>
154.         <SkewTransform/>
155.         <RotateTransform/>
156.         <TranslateTransform X="20" Y="330"/>
157.         </TransformGroup>
158.         </TextBlock.RenderTransform>
159.     </TextBlock>
160.     <TextBox x:Name="x" Text="x" FontSize="20"
RenderTransformOrigin="0.5,0.5" VerticalAlignment="Top"
HorizontalAlignment="Left" Height="30" Width="100">
161.         <TextBox.RenderTransform>
162.         <TransformGroup>
163.         <ScaleTransform/>
164.         <SkewTransform/>
165.         <RotateTransform/>
166.         <TranslateTransform X="400" Y="270"/>
167.         </TransformGroup>
168.         </TextBox.RenderTransform>
169.     </TextBox>
170.     <TextBox x:Name="FZ" Text="Φ3" FontSize="20"
RenderTransformOrigin="0.5,0.5" VerticalAlignment="Top"
HorizontalAlignment="Left" Height="30" Width="100">
171.         <TextBox.RenderTransform>
172.         <TransformGroup>
173.         <ScaleTransform/>
174.         <SkewTransform/>
175.         <RotateTransform/>
176.         <TranslateTransform X="400" Y="340"/>
177.         </TransformGroup>
178.         </TextBox.RenderTransform>
179.     </TextBox>
180.     <TextBlock Text="c = " FontSize="20" VerticalAlignment="Top"
TextWrapping="Wrap" HorizontalAlignment="Left" RenderTransformOrigin="0.5,0.5"
Width="50" Height="40">
181.         <TextBlock.RenderTransform>
182.         <TransformGroup>
183.         <ScaleTransform/>
184.         <SkewTransform/>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

53
RU.17701729.04.15-01 12 01-1

```

185.         <RotateTransform/>
186.         <TranslateTransform X="20" Y="410"/>
187.     </TransformGroup>
188. </TextBlock.RenderTransform>
189. </TextBlock>
190.     <TextBox x:Name="c" Text="-1" FontSize="20"
RenderTransformOrigin="0.5,0.5" VerticalAlignment="Top"
HorizontalAlignment="Left" Height="30" Width="100">
191.         <TextBox.RenderTransform>
192.             <TransformGroup>
193.                 <ScaleTransform/>
194.                 <SkewTransform/>
195.                 <RotateTransform/>
196.                 <TranslateTransform X="90" Y="410"/>
197.             </TransformGroup>
198.         </TextBox.RenderTransform>
199.     </TextBox>
200.     <TextBlock Text="A = " FontSize="20" VerticalAlignment="Top"
TextWrapping="Wrap" HorizontalAlignment="Left" RenderTransformOrigin="0.5,0.5"
Width="50" Height="40">
201.         <TextBlock.RenderTransform>
202.             <TransformGroup>
203.                 <ScaleTransform/>
204.                 <SkewTransform/>
205.                 <RotateTransform/>
206.                 <TranslateTransform X="215" Y="435"/>
207.             </TransformGroup>
208.         </TextBlock.RenderTransform>
209.     </TextBlock>
210.     <TextBox x:Name="_A" Text="-1" FontSize="20"
RenderTransformOrigin="0.5,0.5" VerticalAlignment="Top"
HorizontalAlignment="Left" Height="30" Width="100">
211.         <TextBox.RenderTransform>
212.             <TransformGroup>
213.                 <ScaleTransform/>
214.                 <SkewTransform/>
215.                 <RotateTransform/>
216.                 <TranslateTransform X="265" Y="435"/>
217.             </TransformGroup>
218.         </TextBox.RenderTransform>
219.     </TextBox>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

220.      <TextBlock Text="n = " FontSize="20" VerticalAlignment="Top"
TextWrapping="Wrap" HorizontalAlignment="Left" RenderTransformOrigin="0.5,0.5"
Width="50" Height="40">
221.      <TextBlock.RenderTransform>
222.      <TransformGroup>
223.      <ScaleTransform/>
224.      <SkewTransform/>
225.      <RotateTransform/>
226.      <TranslateTransform X="20" Y="460"/>
227.      </TransformGroup>
228.      </TextBlock.RenderTransform>
229.      </TextBlock>
230.      <TextBox x:Name="n" Text="-1" FontSize="20"
RenderTransformOrigin="0.5,0.5" VerticalAlignment="Top"
HorizontalAlignment="Left" Height="30" Width="100">
231.      <TextBox.RenderTransform>
232.      <TransformGroup>
233.      <ScaleTransform/>
234.      <SkewTransform/>
235.      <RotateTransform/>
236.      <TranslateTransform X="90" Y="460"/>
237.      </TransformGroup>
238.      </TextBox.RenderTransform>
239.      </TextBox>
240.      <TextBlock Text="m = " FontSize="20" TextWrapping="Wrap"
VerticalAlignment="Top" HorizontalAlignment="Left"
RenderTransformOrigin="0.5,0.5" Width="50" Height="40">
241.      <TextBlock.RenderTransform>
242.      <TransformGroup>
243.      <ScaleTransform/>
244.      <SkewTransform/>
245.      <RotateTransform/>
246.      <TranslateTransform X="390" Y="410"/>
247.      </TransformGroup>
248.      </TextBlock.RenderTransform>
249.      </TextBlock>
250.      <TextBox x:Name="m" Text="-1" FontSize="20"
RenderTransformOrigin="0.5,0.5" VerticalAlignment="Top"
HorizontalAlignment="Right" Height="30" Width="100" >
251.      <TextBox.RenderTransform>
252.      <TransformGroup>
253.      <ScaleTransform/>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

55
RU.17701729.04.15-01 12 01-1

```

254.         <SkewTransform/>
255.         <RotateTransform/>
256.         <TranslateTransform X="-25" Y="410"/>
257.     </TransformGroup>
258. </TextBox.RenderTransform>
259. </TextBox>
260.     <TextBlock Text="z = " FontSize="20" TextWrapping="Wrap"
        VerticalAlignment="Top" HorizontalAlignment="Left"
        RenderTransformOrigin="0.5,0.5" Width="50" Height="40">
261.         <TextBlock.RenderTransform>
262.             <TransformGroup>
263.                 <ScaleTransform/>
264.                 <SkewTransform/>
265.                 <RotateTransform/>
266.                 <TranslateTransform X="390" Y="460"/>
267.             </TransformGroup>
268.         </TextBlock.RenderTransform>
269.     </TextBlock>
270.     <TextBox x:Name="z" Text="-1" FontSize="20"
        RenderTransformOrigin="0.5,0.5" VerticalAlignment="Top"
        HorizontalAlignment="Right" Height="30" Width="100">
271.         <TextBox.RenderTransform>
272.             <TransformGroup>
273.                 <ScaleTransform/>
274.                 <SkewTransform/>
275.                 <RotateTransform/>
276.                 <TranslateTransform X="-25" Y="460"/>
277.             </TransformGroup>
278.         </TextBox.RenderTransform>
279.     </TextBox>
280.     <Button Name="Save" VerticalAlignment="Bottom"
        HorizontalAlignment="Center" Content="Сохранить изменения и закрыть окно"
        RenderTransformOrigin="0.5,0.5" Width="240" Height="35" Click="Save_Click" >
281.         <Button.RenderTransform>
282.             <TransformGroup>
283.                 <ScaleTransform/>
284.                 <SkewTransform/>
285.                 <RotateTransform/>
286.                 <TranslateTransform Y="-20"/>
287.             </TransformGroup>
288.         </Button.RenderTransform>
289.     </Button>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата


```
290.         </Grid>
291.     </Window>
```

- **DataSettings.cs**

```
1. using Activision_Mendeleyev_table.HelperClasses;
2. using System.Windows;
3.
4. namespace Activision_Mendeleyev_table
5. {
6.     /// <summary>
7.     /// Логика взаимодействия для DataSettings.xaml
8.     /// </summary>
9.     public partial class DataSettings : Window
10.    {
11.        /// <summary>
12.        /// Система соединений
13.        /// </summary>
14.        private BinSystem sys;
15.
16.        /// <summary>
17.        /// Настройка некоторых параметров системы
18.        /// </summary>
19.        /// <param name="sys">система соединений</param>
20.        public DataSettings(BinSystem sys)
21.        {
22.            InitializeComponent();
23.            this.sys = sys;
24.            if (sys != null)
25.            {
26.                elemA.Text = sys.ElementA;
27.                elemB.Text = sys.ElementB;
28.                elemX.Text = sys.ElementX;
29.                double[] par = sys.GetData();
30.                numX.Text = sys.numX.ToString();
31.                numB.Text = sys.numB.ToString();
32.                numA.Text = sys.numA.ToString();
33.                z.Text = par[3].ToString();
34.                c.Text = par[0].ToString();
35.                m.Text = par[1].ToString();
36.                n.Text = par[2].ToString();
37.                _A.Text = sys.A.ToString();
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

38.         }
39.     }
40.
41.     /// <summary>
42.     /// Возвращает систему соединений
43.     /// </summary>
44.     public BinSystem GetBS() { return sys; }
45.
46.     private void Window_Closing(object sender,
System.ComponentModel.CancelEventArgs e)
47.     {
48.         if (MessageBox.Show("Вы точно хотите закрыть окно? Все
несохраненные данные будут удалены!", "", MessageBoxButton.YesNo,
MessageBoxImage.Question) == MessageBoxResult.No)
49.             e.Cancel = true;
50.     }
51.
52.     /// <summary>
53.     /// Сохраняет параметры в системе и закрывает окно
54.     /// </summary>
55.     private void Save_Click(object sender, RoutedEventArgs e)
56.     {
57.
58.         Composition A = MendeleevTable.Elems.Find(x => x.Name ==
elemA.Text);
59.         Composition B = MendeleevTable.Elems.Find(x => x.Name ==
elemB.Text);
60.         Composition X = MendeleevTable.Elems.Find(x => x.Name ==
elemX.Text);
61.         if (A == null || B == null || X == null)
62.             MessageBox.Show("Неверные заданы названия элементов входящих в
систему! Измените их в меню настроек!", "Error",
63.                 MessageBoxButton.OK, MessageBoxImage.Error);
64.         else
65.         {
66.             sys = new BinSystem(elemA.Text + elemX.Text + '-' + elemB.Text
+ elemX.Text, A, B, X);
67.             sys.symbols[0] = r.Text;
68.             sys.symbols[1] = x.Text;
69.             sys.symbols[2] = FZ.Text;
70.             _A.Text = _A.Text.Replace('.', ',');
71.             double q = -1;

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

72.         int w = -1;
73.         if (!double.TryParse(z.Text, out q) || q < 0)
74.             MessageBox.Show("Поле z - неотрицательное число!", "Error",
                MessageBoxButton.OK, MessageBoxImage.Error);
75.         else if (!int.TryParse(numA.Text, out w) || w < 0)
76.             MessageBox.Show("Поле numA - целое неотрицательное число!",
                "Error", MessageBoxButton.OK, MessageBoxImage.Error);
77.         else if (!int.TryParse(numB.Text, out w) || w < 0)
78.             MessageBox.Show("Поле numB - целое неотрицательное число!",
                "Error", MessageBoxButton.OK, MessageBoxImage.Error);
79.         else if (!int.TryParse(numX.Text, out w) || w < 0)
80.             MessageBox.Show("Поле numX - целое неотрицательное число!",
                "Error", MessageBoxButton.OK, MessageBoxImage.Error);
81.         else if (!double.TryParse(m.Text, out q) || q < 0)
82.             MessageBox.Show("Поле m - неотрицательное число!", "Error",
                MessageBoxButton.OK, MessageBoxImage.Error);
83.         else if (!double.TryParse(c.Text, out q) || q < 0)
84.             MessageBox.Show("Поле c - неотрицательное число!", "Error",
                MessageBoxButton.OK, MessageBoxImage.Error);
85.         else if (!double.TryParse(n.Text, out q) || q < 0)
86.             MessageBox.Show("Поле n - неотрицательное число!", "Error",
                MessageBoxButton.OK, MessageBoxImage.Error);
87.         else if (!double.TryParse(_A.Text, out q) || q < 0)
88.             MessageBox.Show("Поле A - неотрицательное число!", "Error",
                MessageBoxButton.OK, MessageBoxImage.Error);
89.         else
90.         {
91.             sys.SetData(double.Parse(c.Text), double.Parse(m.Text),
                double.Parse(n.Text), double.Parse(z.Text));
92.             sys.numA = int.Parse(numA.Text);
93.             sys.numB = int.Parse(numB.Text);
94.             sys.numX = int.Parse(numX.Text);
95.             sys.A = double.Parse(_A.Text);
96.             this.Close();
97.         }
98.     }
99. }
100. }
101. }
102.

```

9. Класс DomeOfDecay:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

- **DomeOfDecay.xaml**

```

1. <Window x:Class="Activision_Mendeleyev_table.DomeOfDecay"
2.     xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
3.     xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
4.     xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
5.     xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
6.     xmlns:xctk="http://schemas.xceed.com/wpf/xaml/toolkit"
7.     xmlns:wfi="clr-
    namespace:System.Windows.Forms.Integration;assembly=WindowsFormsIntegration"
8.     xmlns:local="clr-namespace:Activision_Mendeleyev_table.HelperClasses"
9.     mc:Ignorable="d"
10.     Icon="icon.jpeg"
11.     Title="Купол распада" Height="465" Width="557"
    WindowStartupLocation="CenterScreen" MinHeight="765" MinWidth="857"
    Closing="Window_Closing" SizeChanged="Window_SizeChanged"
    SourceInitialized="Window_SourceInitialized">
12.     <Window.Resources>
13.         <local:RoundConverter x:Key="RoundConverter"/></local:RoundConverter>
14.     </Window.Resources>
15.     <Grid>
16.         <Grid.RowDefinitions>
17.             <RowDefinition Height="20"/>
18.             <RowDefinition Height="*" />
19.             <RowDefinition Height="50"/>
20.         </Grid.RowDefinitions>
21.         <Grid.ColumnDefinitions>
22.             <ColumnDefinition Width="21*" />
23.             <ColumnDefinition Width="628*" />
24.             <ColumnDefinition Width="200" />
25.         </Grid.ColumnDefinitions>
26.         <Menu Grid.Row="0" Grid.ColumnSpan="2">
27.             <MenuItem Header="Графическое меню">
28.                 <MenuItem Header="Настройка цвета">
29.                     <MenuItem Header="Теория">
30.                         <xctk:ColorPicker Name="Theory" Width="50"
    SelectedColor="Black" ShowTabHeaders="False" />
31.                     </MenuItem>
32.                     <MenuItem Header="Эксперимент">
33.                         <xctk:ColorPicker Name="Experiment" Width="50"
    SelectedColor="Red" ShowTabHeaders="False" />
34.                     </MenuItem>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

35.             <MenuItem Header="Аппроксимация/чувствительность">
36.                 <xctk:ColorPicker Name="Approximation" Width="50"
SelectedColor="Green" ShowTabHeaders="False" />
37.             </MenuItem>
38.         </MenuItem>
39.         <MenuItem Header="Настройка границ">
40.             <MenuItem Header="Верхняя температура" >
41.                 <TextBox Name="UpT" Text="-1" Width="50" />
42.             </MenuItem>
43.             <MenuItem Header="Нижняя температура" >
44.                 <TextBox Name="DownT" Text="-1" Width="50" />
45.             </MenuItem>
46.             <MenuItem Header="Верхняя граница параметра Rmin" >
47.                 <TextBox Name="UpR" Text="10" Width="50"
IsEnabled="False"/>
48.             </MenuItem>
49.             <MenuItem Header="Нижняя граница параметра Rmin" >
50.                 <TextBox Name="DownR" Text="0.01" Width="50"
IsEnabled="False" />
51.             </MenuItem>
52.             <MenuItem Header="Верхняя граница параметра delEps" >
53.                 <TextBox Name="UpdE" Text="1" Width="50"
IsEnabled="False" />
54.             </MenuItem>
55.             <MenuItem Header="Нижняя граница параметра delEps" >
56.                 <TextBox Name="DowndE" Text="0.01" Width="50"
IsEnabled="False" />
57.             </MenuItem>
58.             <MenuItem Header="Верхняя граница параметра c" >
59.                 <TextBox Name="Upc" Text="100" Width="50"
IsEnabled="False" />
60.             </MenuItem>
61.             <MenuItem Header="Нижняя граница параметра c" >
62.                 <TextBox Name="Downc" Text="0.01" Width="50"
IsEnabled="False" />
63.             </MenuItem>
64.         </MenuItem>
65.         <MenuItem Name="IsExpPoints" Header="Эксперимент точками"
IsCheckable="True" IsChecked="True" Checked="MenuItem_Checked"
Unchecked="MenuItem_Unchecked" />
66.     </MenuItem>
67.     <MenuItem Header="Настройка данных" Click="DataSettings_Click" />

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

61
RU.17701729.04.15-01 12 01-1

```

68.         <MenuItem Header="Оценка чувствительности"
Click="Sensitivity_Click" />
69.     </Menu>
70.     <wf:WindowsFormsHost Name="host" Grid.Column="0" Grid.Row="1"
Grid.ColumnSpan="2"/>
71.     <DataGrid Name="Points" Grid.Column="2" Grid.Row="1"
AutoGenerateColumns="False" CanUserReorderColumns="False"
CanUserSortColumns="False" CanUserResizeRows="False"
CanUserResizeColumns="False" CanUserAddRows="True" CanUserDeleteRows="True"
ColumnWidth="*" CellEditEnding="Points_CellEditEnding"
RowEditEnding="Points_RowEditEnding"/>
72.     <Label Name="dE_label" Content="delEps" Grid.Column="2" Grid.Row="1"
HorizontalAlignment="Center" Visibility="Hidden" VerticalAlignment="Top"
Margin="78,0" />
73.     <Label Name="R_label" Content="R" Grid.Column="2" Grid.Row="1"
HorizontalAlignment="Left" VerticalAlignment="Top" Visibility="Hidden"
RenderTransformOrigin="0.5,0.5" >
74.         <Label.RenderTransform>
75.             <TransformGroup>
76.                 <ScaleTransform/>
77.                 <SkewTransform/>
78.                 <RotateTransform/>
79.                 <TranslateTransform X="30"/>
80.             </TransformGroup>
81.         </Label.RenderTransform>
82.     </Label>
83.     <Label Name="c_label" Content="c" Grid.Column="2" Grid.Row="1"
HorizontalAlignment="Right" VerticalAlignment="Top" Visibility="Hidden"
RenderTransformOrigin="0.5,0.5" >
84.         <Label.RenderTransform>
85.             <TransformGroup>
86.                 <ScaleTransform/>
87.                 <SkewTransform/>
88.                 <RotateTransform/>
89.                 <TranslateTransform X="-20"/>
90.             </TransformGroup>
91.         </Label.RenderTransform>
92.     </Label>
93.     <TextBox Name="dE_text" Grid.Column="2" Grid.Row="1" Text="{Binding
ElementName=dE, Path=Value, UpdateSourceTrigger=PropertyChanged,
Converter={StaticResource RoundConverter}}" HorizontalAlignment="Center"

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

Visibility="Hidden" VerticalAlignment="Top" Height="20" Width="50"
IsReadOnly="True" RenderTransformOrigin="0.5,0.5">
94.         <TextBox.RenderTransform>
95.             <TransformGroup>
96.                 <ScaleTransform/>
97.                 <SkewTransform/>
98.                 <RotateTransform/>
99.                 <TranslateTransform Y="35"/>
100.            </TransformGroup>
101.        </TextBox.RenderTransform>
102.    </TextBox>
103.    <TextBox Name="R_text" Grid.Column="2" Grid.Row="1"
        Text="{Binding ElementName=R, Path=Value, UpdateSourceTrigger=PropertyChanged,
        Converter={StaticResource RoundConverter}}" HorizontalAlignment="Left"
        VerticalAlignment="Top" Visibility="Hidden" Height="20" Width="50"
        RenderTransformOrigin="0.5,0.5" IsReadOnly="True">
104.        <TextBox.RenderTransform>
105.            <TransformGroup>
106.                <ScaleTransform/>
107.                <SkewTransform/>
108.                <RotateTransform/>
109.                <TranslateTransform X="20" Y="35"/>
110.            </TransformGroup>
111.        </TextBox.RenderTransform>
112.    </TextBox>
113.    <TextBox Name="c_text" Grid.Column="2" Grid.Row="1"
        Text="{Binding ElementName=c, Path=Value, UpdateSourceTrigger=PropertyChanged,
        Converter={StaticResource RoundConverter}}" HorizontalAlignment="Right"
        VerticalAlignment="Top" Visibility="Hidden" Height="20" Width="50"
        RenderTransformOrigin="0.5,0.5" IsReadOnly="True">
114.        <TextBox.RenderTransform>
115.            <TransformGroup>
116.                <ScaleTransform/>
117.                <SkewTransform/>
118.                <RotateTransform/>
119.                <TranslateTransform X="-20" Y="35"/>
120.            </TransformGroup>
121.        </TextBox.RenderTransform>
122.    </TextBox>
123.    <Slider Name="R" Grid.Column="2" Grid.Row="1"
        Orientation="Vertical" HorizontalAlignment="Left" VerticalAlignment="Top"

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

Height="600" Visibility="Hidden" RenderTransformOrigin="0.5,0.5"
ValueChanged="R_ValueChanged">
124.         <Slider.RenderTransform>
125.             <TransformGroup>
126.                 <ScaleTransform/>
127.                 <SkewTransform/>
128.                 <RotateTransform/>
129.                 <TranslateTransform X="20" Y="70"/>
130.             </TransformGroup>
131.         </Slider.RenderTransform>
132.     </Slider>
133.     <Slider Name="dE" Grid.Column="2" Grid.Row="1"
Orientation="Vertical" HorizontalAlignment="Center" VerticalAlignment="Top"
Height="600" Visibility="Hidden" RenderTransformOrigin="0.5,0.5"
ValueChanged="dE_ValueChanged">
134.         <Slider.RenderTransform>
135.             <TransformGroup>
136.                 <ScaleTransform/>
137.                 <SkewTransform/>
138.                 <RotateTransform/>
139.                 <TranslateTransform Y="70"/>
140.             </TransformGroup>
141.         </Slider.RenderTransform>
142.     </Slider>
143.     <Slider Name="c" Grid.Column="2" Grid.Row="1"
Orientation="Vertical" HorizontalAlignment="Right" VerticalAlignment="Top"
Height="600" Visibility="Hidden" RenderTransformOrigin="0.5,0.5"
ValueChanged="c_ValueChanged">
144.         <Slider.RenderTransform>
145.             <TransformGroup>
146.                 <ScaleTransform/>
147.                 <SkewTransform/>
148.                 <RotateTransform/>
149.                 <TranslateTransform X="-20" Y="70"/>
150.             </TransformGroup>
151.         </Slider.RenderTransform>
152.     </Slider>
153.     <Button Name="Build" Grid.Column="0" Grid.Row="2"
Content="Построить купол" HorizontalAlignment="Left" VerticalAlignment="Center"
Width="150" Height="34" RenderTransformOrigin="0.5,0.5" Click="Build_Click"
Grid.ColumnSpan="2" Margin="0,8">
154.         <Button.RenderTransform>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата


```

155.                <TransformGroup>
156.                    <ScaleTransform/>
157.                    <SkewTransform/>
158.                    <RotateTransform/>
159.                    <TranslateTransform X="20"/>
160.                </TransformGroup>
161.            </Button.RenderTransform>
162.        </Button>
163.        <Button Name="Approxim" Grid.Column="0" Grid.Row="2"
Content="Построить функцию Hsm и аппроксимировать" HorizontalAlignment="Left"
VerticalAlignment="Center" Width="285" Height="34"
RenderTransformOrigin="0.5,0.5" Click="Approxim_Click" Grid.ColumnSpan="2"
Margin="0,8">
164.            <Button.RenderTransform>
165.                <TransformGroup>
166.                    <ScaleTransform/>
167.                    <SkewTransform/>
168.                    <RotateTransform/>
169.                    <TranslateTransform X="190"/>
170.                </TransformGroup>
171.            </Button.RenderTransform>
172.        </Button>
173.        <Button Name="Back" Grid.Column="0" Grid.Row="2" Content="Назад"
HorizontalAlignment="Center" VerticalAlignment="Center" Width="100" Height="34"
Click="Back_Click" Grid.ColumnSpan="2" Visibility="Hidden" />
174.        <Button Name="Save" Grid.Column="1" Grid.Row="2"
Content="Сохранить точки" HorizontalAlignment="Right"
VerticalAlignment="Center" Height="34" Width="150" Click="Save_Click"
RenderTransformOrigin="0.5,0.5" Margin="0,8">
175.            <Button.RenderTransform>
176.                <TransformGroup>
177.                    <ScaleTransform/>
178.                    <SkewTransform/>
179.                    <RotateTransform/>
180.                    <TranslateTransform X="10"/>
181.                </TransformGroup>
182.            </Button.RenderTransform>
183.        </Button>
184.        <Button Name="Load" Grid.Column="2" Grid.Row="2"
Content="Загрузить точки" HorizontalAlignment="Right"
VerticalAlignment="Center" Width="150" Height="34"
RenderTransformOrigin="0.5,0.5" Click="Load_Click" Margin="0,8">

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

185.             <Button.RenderTransform>
186.                 <TransformGroup>
187.                     <ScaleTransform/>
188.                     <SkewTransform/>
189.                     <RotateTransform/>
190.                     <TranslateTransform X="-20"/>
191.                 </TransformGroup>
192.             </Button.RenderTransform>
193.         </Button>
194.     </Grid>
195. </Window>

```

- **DomeOfDecay.cs**

```

1. using Activision_Mendeleyev_table.Approximation;
2. using Activision_Mendeleyev_table.HelperClasses;
3. using System;
4. using System.Collections.Generic;
5. using System.Drawing;
6. using System.IO;
7. using System.Runtime.InteropServices;
8. using System.Text.RegularExpressions;
9. using System.Windows;
10. using System.Windows.Controls;
11. using System.Windows.Data;
12.
13. namespace Activision_Mendeleyev_table
14. {
15.     /// <summary>
16.     /// Логика взаимодействия для DomeOfDecay.xaml
17.     /// </summary>
18.     public partial class DomeOfDecay : Window
19.     {
20.         /// <summary>
21.         /// Лист точек, представленных в DataGrid
22.         /// </summary>
23.         private List<List<double>>> dat = new List<List<double>>>();
24.         /// <summary>
25.         /// Системы соединений (базовая и аппроксимированная)
26.         /// </summary>
27.         private BinSystem sys, sys_ap = null;
28.         /// <summary>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

29.      /// Графики(купол распада/функция смещения и аппроксимированная функция
    смещения)
30.      /// </summary>
31.      private DrawingClasses.CollapseGraph graph, graph_ap;
32.      /// <summary>
33.      /// Флаг: true - купол распада, false - функция смещения
34.      /// </summary>
35.      private bool f = true;
36.      private System.Windows.Forms.PictureBox diag = new
    System.Windows.Forms.PictureBox();
37.
38.      [DllImport("user32.dll")]
39.      private static extern int GetWindowLong(IntPtr hWnd, int nIndex);
40.      [DllImport("user32.dll")]
41.      private static extern int SetWindowLong(IntPtr hWnd, int nIndex, int
    dwNewLong);
42.
43.      private const int GWL_STYLE = -16;
44.      private const int WS_MAXIMIZEBOX = 0x10000;
45.
46.      /// <summary>
47.      /// Инициализация элемента host
48.      /// </summary>
49.      private void Window_SourceInitialized(object sender, EventArgs e)
50.      {
51.          var hwnd = new
    System.Windows.Interop.WindowInteropHelper((Window)sender).Handle;
52.          var value = GetWindowLong(hwnd, GWL_STYLE);
53.          SetWindowLong(hwnd, GWL_STYLE, (int)(value & ~WS_MAXIMIZEBOX));
54.      }
55.
56.      /// <summary>
57.      /// Первоначальные настройки и построение купола распада
58.      /// </summary>
59.      /// <param name="name">обозначение системы</param>
60.      public DomeOfDecay(string name)
61.      {
62.          InitializeComponent();
63.
64.          //Привязка горячей клавише Delete к методу DeleteSelectedRows
65.          new HotKey(System.Windows.Input.Key.Delete, KeyModifier.None,
    DeleteSelectedRows);

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

66.         string[] elems = Parse(name);
67.         Composition A = MendeleevTable.Elems.Find(x => x.Name == elems[0]);
68.         Composition B = MendeleevTable.Elems.Find(x => x.Name == elems[1]);
69.         Composition X = MendeleevTable.Elems.Find(x => x.Name == elems[2]);
70.         if (A == null || B == null || X == null)
71.             MessageBox.Show("Неверно заданы названия элементов входящих в
систему! Измените их в меню настроек!", "Error",
72.                             MessageBoxButtons.OK, MessageBoxIcon.Error);
73.         else
74.             sys = new BinSystem(name, A, B, X);
75.
76.         DataSettings ds = new DataSettings(sys);
77.         ds.ShowDialog();
78.         sys = ds.GetBS();
79.
80.         host.Child = diag;
81.         diag.Paint += new
System.Windows.Forms.PaintEventHandler(diag_Paint);
82.
83.         Points.Columns.Add(new DataGridViewTextBoxColumn()
84.         {
85.             Header = "x",
86.             Binding = new Binding("[0]")
87.         });
88.         Points.Columns.Add(new DataGridViewTextBoxColumn()
89.         {
90.             Header = "y",
91.             Binding = new Binding("[1]")
92.         });
93.         Points.ItemSource = dat;
94.     }
95.
96.     /// <summary>
97.     /// Получение химических элементов из обозначения системы
98.     /// </summary>
99.     /// <param name="s">обозначение системы</param>
100.    /// <returns>массив обозначений химических элементов</returns>
101.    public static string[] Parse(string s)
102.    {
103.        string[] names = new string[] { "", "", "" };
104.
105.        s = s.Replace(" ", "");

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

106.
107.         if (new Regex(@"[A-Z]{1}[a-z]?[_0123456789]*[A-Z]{1}[a-
           z]?[_0123456789]*[-]{1}[A-Z]{1}[a-z]?[_0123456789]*[A-Z]{1}[a-
           z]?[_0123456789]*").IsMatch(s))
108.         {
109.             for (int i = 0; i < 3; i++)
110.             {
111.                 Regex myReg = new Regex(@"[A-Z]{1}[a-z]?"); //шаблон
элементa
112.                 Match match = myReg.Match(s);
113.
114.                 s = s.Replace(match.Value, "");
115.                 names[i] = match.Value;
116.             }
117.
118.             // поставка элемента X на третье место в массиве
119.             string z = names[2];
120.             names[2] = names[1];
121.             names[1] = z;
122.         }
123.
124.         return names;
125.     }
126.
127.     private void Window_Closing(object sender,
System.ComponentModel.CancelEventArgs e)
128.     {
129.         if (MessageBox.Show("Вы точно хотите закрыть окно? Все
несохраненные данные будут удалены!", "", MessageBoxButton.YesNo,
MessageBoxImage.Question) == MessageBoxResult.No)
130.             e.Cancel = true;
131.         else
132.             DrawingClasses.CollapseGraph.ClearExperiment();
133.     }
134.
135.     private void Window_SizeChanged(object sender,
SizeChangedEventArgs e)
136.     {
137.         MaxWidth = e.NewSize.Height + 92;
138.         Width = e.NewSize.Height + 92;
139.     }
140.

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

141.          /// <summary>
142.          /// Построение графиков
143.          /// </summary>
144.          private void diag_Paint(object sender,
            System.Windows.Forms.PaintEventArgs e)
145.          {
146.              Graphics g = e.Graphics;
147.              g.SmoothingMode =
            System.Drawing.Drawing2D.SmoothingMode.HighQuality;
148.
149.              try
150.              {
151.                  graph = new DrawingClasses.CollapseGraph(g, sys,
            diag.Width);
152.
153.                  if (f)
154.                      graph.DrawCollapse();
155.                  else
156.                  {
157.                      if (sys_ap != null)
158.                      {
159.                          //MessageBox.Show(sys_ap.Tmax.ToString());
160.                          graph_ap = new DrawingClasses.CollapseGraph(g,
            sys_ap, diag.Width);
161.                          graph_ap.DrawDH();
162.                      }
163.                      //else
164.                          // MessageBox.Show(sys.Tmax.ToString());
165.                      graph.DrawDH(false);
166.                  }
167.
168.                  graph.DrawAxes();
169.                  graph.DrawExperiment();
170.              }
171.              catch (Exception ex)
172.              {
173.                  if (ex.InnerException != null &&
            ex.InnerException.Message == "MyException")
174.                      MessageBox.Show(ex.Message, "Error",
            MessageBoxButtons.OK, MessageBoxIcon.Error);
175.                  else

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

176.                MessageBox.Show("Неверные данные для построения
                купола! Измените их в таблицах или в меню настроек!", "Error",
                MessageBoxButton.OK, MessageBoxImage.Error);
177.                }
178.            }
179.
180.            private void Points_CellEditEnding(object sender,
                DataGridViewCellEditEndingEventArgs e)
181.            {
182.                (e.EditingElement as TextBox).Text = (e.EditingElement as
                TextBox).Text.Replace(',', '.', '.');
183.                if (dat[e.Row.GetIndex()].Capacity == 0)
184.                {
185.                    dat[e.Row.GetIndex()].Add(0);
186.                    dat[e.Row.GetIndex()].Add(0);
187.                }
188.                if (!float.TryParse((e.EditingElement as
                TextBox).Text.Replace('.', ','), out float p) || p < 0)
189.                {
190.                    MessageBox.Show("Координаты точки должны быть
                неотрицательным числом!", "Error", MessageBoxButton.OK, MessageBoxImage.Error);
191.                    (e.EditingElement as TextBox).Text = "";
192.
193.                    e.Cancel = true;
194.                }
195.                if (e.Column.DisplayIndex == 0)
196.                    dat[e.Row.GetIndex()][0] = p;
197.                else
198.                    dat[e.Row.GetIndex()][1] = p;
199.            }
200.
201.            /// <summary>
202.            /// Построение купола распада
203.            /// </summary>
204.            private void Build_Click(object sender, RoutedEventArgs e)
205.            {
206.                if (!f && sys_ap != null)
207.                if (MessageBox.Show("Использовать при построении купола
                новые значения параметров?", "", MessageBoxButton.YesNo,
                MessageBoxImage.Question) == MessageBoxResult.Yes)
208.                    sys = sys_ap.Clone();
209.

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

210.         f = true;
211.
212.         SetColor();
213.         SetBorders();
214.
215.         diag.Refresh();
216.     }
217.
218.     /// <summary>
219.     /// Запускает оценку чувствительности
220.     /// </summary>
221.     private void Sensitivity_Click(object sender, RoutedEventArgs e)
222.     {
223.         f = false;
224.         sys_ap = sys.Clone();
225.         Points.Visibility = Visibility.Hidden;
226.         Build.Visibility = Visibility.Hidden;
227.         Approx.Visibility = Visibility.Hidden;
228.         Save.Visibility = Visibility.Hidden;
229.         Load.Visibility = Visibility.Hidden;
230.         R.Visibility = Visibility.Visible;
231.         dE.Visibility = Visibility.Visible;
232.         c.Visibility = Visibility.Visible;
233.         R_label.Visibility = Visibility.Visible;
234.         dE_label.Visibility = Visibility.Visible;
235.         c_label.Visibility = Visibility.Visible;
236.         R_text.Visibility = Visibility.Visible;
237.         dE_text.Visibility = Visibility.Visible;
238.         c_text.Visibility = Visibility.Visible;
239.         Back.Visibility = Visibility.Visible;
240.         DownR.IsEnabled = true;
241.         UpdE.IsEnabled = true;
242.         Downc.IsEnabled = true;
243.         UpR.IsEnabled = true;
244.         DowndE.IsEnabled = true;
245.         Upc.IsEnabled = true;
246.         IsExpPoints.IsEnabled = false;
247.
248.         R.Value = Math.Min(sys_ap.R(0), sys_ap.R(1));
249.         c.Value = sys_ap.GetData()[0];
250.         dE.Value = sys_ap.delEps;
251.

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата


```

252.         SetColor();
253.         SetBorders();
254.
255.         diag.Refresh();
256.     }
257.
258.     /// <summary>
259.     /// Сохранение экспериментальных точек в файл
260.     /// </summary>
261.     private void Save_Click(object sender, RoutedEventArgs e)
262.     {
263.         Microsoft.Win32.SaveFileDialog dlg = new
Microsoft.Win32.SaveFileDialog
264.         {
265.             FileName = "Points",
266.             DefaultExt = ".txt",
267.             Filter = "Text files (.txt)|*.txt"
268.         };
269.
270.
271.         if (dlg.ShowDialog() == true)
272.             using (FileStream fs = new FileStream(dlg.FileName,
                FileMode.Create, FileAccess.Write))
273.                 using (StreamWriter sw = new StreamWriter(fs))
274.                     for (int i = 0; i < dat.Count; i++)
275.                         sw.WriteLine(dat[i][0] + " " + dat[i][1]);
276.     }
277.
278.     /// <summary>
279.     /// Загрузка экспериментальных точек из файла
280.     /// </summary>
281.     private void Load_Click(object sender, RoutedEventArgs e)
282.     {
283.         Microsoft.Win32.OpenFileDialog dlg = new
Microsoft.Win32.OpenFileDialog
284.         {
285.             Filter = "Text files (.txt)|*.txt",
286.             CheckFileExists = true
287.         };
288.
289.         try
290.         {

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

291.         if (dlg.ShowDialog() == true)
292.         {
293.             dat.Clear();
294.             using (FileStream fs = new FileStream(dlg.FileName,
                FileMode.Open, FileAccess.Read))
295.             using (StreamReader sr = new StreamReader(fs))
296.                 while (!sr.EndOfStream)
297.                 {
298.                     string s = sr.ReadLine();
299.                     dat.Add(new List<double> {
                double.Parse(s.Split()[0]), double.Parse(s.Split()[1]) });
300.                 }
301.             }
302.         }
303.         catch (Exception)
304.         {
305.             MessageBox.Show("Неверный формат файла!");
306.         }
307.         Points.Items.Refresh();
308.         Points_RowEditEnding(this, new
                DataGridViewEditEventArgs(new DataGridView(), DataGridViewEditAction.Commit));
309.     }
310.
311.     /// <summary>
312.     /// Открывает окно настройки параметров
313.     /// </summary>
314.     private void DataSettings_Click(object sender, RoutedEventArgs e)
315.     {
316.         DataSettings ds = new DataSettings(sys);
317.         ds.ShowDialog();
318.         sys = ds.GetBS();
319.
320.         diag.Refresh();
321.     }
322.
323.     private void Points_RowEditEnding(object sender,
                DataGridViewEditEventArgs e)
324.     {
325.         DrawingClasses.CollapseGraph.ClearExperiment();
326.         for (int i = 0; i < dat.Count; i++)
327.             DrawingClasses.CollapseGraph.AddExperimentalPoint(dat[i][0], dat[i][1]);

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

328.             diag.Refresh();
329.         }
330.
331.         /// <summary>
332.         /// Удаляет выделенные строки из таблицы
333.         /// </summary>
334.         /// <param name="hotKey">горячая клавиша для вызова
           метода</param>
335.         private void DeleteSelectedRows(HotKey hotKey)
336.         {
337.             try
338.             {
339.                 while (Points.SelectedItems.Count > 0)
340.                 {
341.                     int selectedIndex = Points.SelectedIndex;
342.
343.                     DrawingClasses.CollapseGraph.RemoveSelectedPoint(selectedIndex);
344.                     (Points.ItemsSource as
           List<List<double>>>).RemoveAt(selectedIndex);
345.                     Points.Items.Refresh();
346.                 }
347.                 diag.Refresh();
348.             }
349.             catch (Exception)
350.             {
351.                 MessageBox.Show("Невозможно удалить этот элемент!",
           "Error", MessageBoxButton.OK, MessageBoxImage.Error);
352.             }
353.
354.         /// <summary>
355.         /// Задает флаг, определяющий формат отображения эксперимента
           точками
356.         /// </summary>
357.         private void MenuItem_Checked(object sender, RoutedEventArgs e)
358.         {
359.             DrawingClasses.CollapseGraph.ExperimentIsPoints = true;
360.         }
361.         /// <summary>
362.         /// Задает флаг, определяющий формат отображения эксперимента
           ломанными
363.         /// </summary>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

364.         private void MenuItem_Unchecked(object sender, RoutedEventArgs e)
365.         {
366.             DrawingClasses.CollapseGraph.ExperimentIsPoints = false;
367.         }
368.
369.         /// <summary>
370.         /// Запускает аппроксимацию функции смещения
371.         /// </summary>
372.         private void Approx_Click(object sender, RoutedEventArgs e)
373.         {
374.             f = false;
375.
376.             SetColor();
377.             SetBorders();
378.
379.             if (sys != null)
380.                 Approximate(new double[] { Math.Min(sys.R(1), sys.R(0)),
381. sys.delEps, sys.GetData()[0] });
382.
383.             diag.Refresh();
384.         }
385.         /// <summary>
386.         /// Задаёт границы параметров
387.         /// </summary>
388.         private void SetBorders()
389.         {
390.             int t = -1;
391.             if (!int.TryParse(DownT.Text, out t))
392.                 MessageBox.Show("Неправильно установлено нижняя граница
393. температуры!", "Error", MessageBoxButton.OK, MessageBoxImage.Error);
394.             DrawingClasses.CollapseGraph.DownTemp = t;
395.
396.             t = -1;
397.             if (!int.TryParse(UpT.Text, out t))
398.                 MessageBox.Show("Неправильно установлено верхняя граница
399. температуры!", "Error", MessageBoxButton.OK, MessageBoxImage.Error);
400.             DrawingClasses.CollapseGraph.UpTemp = t;
401.
402.             if (!f && sys_ap != null)
403.             {

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

402.                                     if (!double.TryParse(UpR.Text.Replace('.', ','), out
double b) && b <= 0)
403.                                     MessageBox.Show("Неправильно установлено верхняя
граница параметра R!", "Error", MessageBoxButton.OK, MessageBoxImage.Error);
404.                                     R.Maximum = b;
405.
406.                                     b = 0.01;
407.                                     if (!double.TryParse(DownR.Text.Replace('.', ','), out b)
&& b <= 0)
408.                                     MessageBox.Show("Неправильно установлено нижняя
граница параметра R!", "Error", MessageBoxButton.OK, MessageBoxImage.Error);
409.                                     R.Minimum = b;
410.
411.                                     b = 0.01;
412.                                     if (!double.TryParse(Upс.Text.Replace('.', ','), out b)
&& b <= 0)
413.                                     MessageBox.Show("Неправильно установлено верхняя
граница параметра с!", "Error", MessageBoxButton.OK, MessageBoxImage.Error);
414.                                     с.Maximum = b;
415.
416.                                     b = 0.01;
417.                                     if (!double.TryParse(Downс.Text.Replace('.', ','), out b)
&& b <= 0)
418.                                     MessageBox.Show("Неправильно установлено нижняя
граница параметра с!", "Error", MessageBoxButton.OK, MessageBoxImage.Error);
419.                                     с.Minimum = b;
420.
421.                                     b = 0.01;
422.                                     if (!double.TryParse(UpdE.Text.Replace('.', ','), out b)
&& b <= 0)
423.                                     MessageBox.Show("Неправильно установлено верхняя
граница параметра dE!", "Error", MessageBoxButton.OK, MessageBoxImage.Error);
424.                                     dE.Maximum = b;
425.
426.                                     b = 0.01;
427.                                     if (!double.TryParse(DowndE.Text.Replace('.', ','), out
b) && b <= 0)
428.                                     MessageBox.Show("Неправильно установлено нижняя
граница параметра dE!", "Error", MessageBoxButton.OK, MessageBoxImage.Error);
429.                                     dE.Minimum = b;
430.                                     }
431.                                     }

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

432.
433.         /// <summary>
434.         /// Задаёт цвета отображения графиков
435.         /// </summary>
436.         private void SetColor()
437.         {
438.             byte[] bytes =
439.                 BitConverter.GetBytes(Convert.ToInt64(Experiment.SelectedColor.Value.B *
440.                 (Math.Pow(256, 0)) +
441.                 Experiment.SelectedColor.Value.G * (Math.Pow(256, 1)) +
442.                 Experiment.SelectedColor.Value.R * (Math.Pow(256, 2))));
443.             DrawingClasses.CollapseGraph.ExperimentColor =
444.                 Color.FromArgb(255, bytes[2], bytes[1], bytes[0]);
445.
446.             bytes =
447.                 BitConverter.GetBytes(Convert.ToInt64(Theory.SelectedColor.Value.B *
448.                 (Math.Pow(256, 0)) +
449.                 Theory.SelectedColor.Value.G * (Math.Pow(256, 1)) +
450.                 Theory.SelectedColor.Value.R * (Math.Pow(256, 2))));
451.             DrawingClasses.CollapseGraph.Color = Color.FromArgb(255,
452.                 bytes[2], bytes[1], bytes[0]);
453.
454.             bytes =
455.                 BitConverter.GetBytes(Convert.ToInt64(Approximation.SelectedColor.Value.B *
456.                 (Math.Pow(256, 0)) +
457.                 Approximation.SelectedColor.Value.G * (Math.Pow(256, 1))
458.                 + Approximation.SelectedColor.Value.R * (Math.Pow(256, 2))));
459.             DrawingClasses.CollapseGraph.ApproximationColor =
460.                 Color.FromArgb(255, bytes[2], bytes[1], bytes[0]);
461.         }
462.
463.         private void c_ValueChanged(object sender,
464.             RoutedPropertyChangedEventArgs<double> e)
465.         {
466.             double[] dat = sys_ap.GetData();
467.             sys_ap.SetData(c.Value, dat[1], dat[2], dat[3]);
468.
469.             SetColor();
470.             SetBorders();
471.
472.             diag.Refresh();
473.         }

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

461.
462.         private void dE_ValueChanged(object sender,
           RoutedPropertyChangedEventArgs<double> e)
463.         {
464.             sys_ap.delEps = dE.Value;
465.
466.             SetColor();
467.             SetBorders();
468.
469.             diag.Refresh();
470.         }
471.
472.         private void R_ValueChanged(object sender,
           RoutedPropertyChangedEventArgs<double> e)
473.         {
474.             sys_ap.R_const = R.Value;
475.
476.             SetColor();
477.             SetBorders();
478.
479.             diag.Refresh();
480.         }
481.
482.         /// <summary>
483.         /// Возвращает к построению купола распада
484.         /// </summary>
485.         private void Back_Click(object sender, RoutedEventArgs e)
486.         {
487.             f = true;
488.             sys_ap = null;
489.             Points.Visibility = Visibility.Visible;
490.             Build.Visibility = Visibility.Visible;
491.             Approxi.Visibility = Visibility.Visible;
492.             Save.Visibility = Visibility.Visible;
493.             Load.Visibility = Visibility.Visible;
494.             R.Visibility = Visibility.Hidden;
495.             dE.Visibility = Visibility.Hidden;
496.             c.Visibility = Visibility.Hidden;
497.             R_label.Visibility = Visibility.Hidden;
498.             dE_label.Visibility = Visibility.Hidden;
499.             c_label.Visibility = Visibility.Hidden;
500.             R_text.Visibility = Visibility.Hidden;

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

501.         dE_text.Visibility = Visibility.Hidden;
502.         c_text.Visibility = Visibility.Hidden;
503.         Back.Visibility = Visibility.Hidden;
504.         DownR.IsEnabled = false;
505.         UpdE.IsEnabled = false;
506.         Downc.IsEnabled = false;
507.         UpR.IsEnabled = false;
508.         DowndE.IsEnabled = false;
509.         Upc.IsEnabled = false;
510.         IsExpPoints.IsEnabled = true;
511.
512.         SetColor();
513.         SetBorders();
514.
515.         diag.Refresh();
516.     }
517.
518.     /// <summary>
519.     ///   Аппроксимация функции
520.     /// </summary>
521.     /// <param name="par">набор изменяемых параметров</param>
522.     private void Approximate(double[] par)
523.     {
524.         List<HelperClasses.Point> Dots = new
525.         List<HelperClasses.Point>();
526.         foreach (List<double> point in dat)
527.             Dots.Add(new HelperClasses.Point(point[0], point[1]));
528.         double[] data = sys.GetData();
529.         Func<double, double[], double> Function = new Func<double,
530.         double[], double>((double x, double[] PP)
531.             => 1000 * x * (1 - x) * ((332 * sys.A / PP[0] * PP[1] * PP[1]
532.             + PP[2] * data[1] * data[2] * data[3] * sys.zX *
533.             (Math.Abs(sys.r1 - sys.r2) / PP[0] * Math.Abs(sys.r1 -
534.             sys.r2) / PP[0]))));
535.         try
536.         {
537.             double[] par_ap = Library.AproxiTab(Dots, Function, par,
538.             Criterion.Criterion_CK0); //TODO min of max and sko
539.             MessageBox.Show(String.Format("R_min = {0:f4}; delta E =
540.             {1:f4}; c = {2:f4}", par_ap[0], par_ap[1], par_ap[2]));

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата


```

537.             sys_ap = sys.Clone();
538.             sys_ap.R_const = par_ap[0];
539.             sys_ap.delEps = par_ap[1];
540.             sys_ap.SetData(par_ap[2], data[1], data[2], data[3]);
541.         }
542.         catch (ArgumentNullException)
543.         {
544.             MessageBox.Show("Точки не заданы! Аппроксимация
невозможна!", "Warning", MessageBoxButton.OK, MessageBoxImage.Warning);
545.         }
546.     }
547. }
548. }
549.

```

10. Класс Help:

- **Help.xaml**

```

1. <Window x:Class="Activision_Mendeleyev_table.Help"
2.         xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
3.         xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
4.         xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
5.         xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
6.         mc:Ignorable="d"
7.         Icon="icon.jpeg"
8.         Title="Help" Height="450" Width="800"
        WindowStartupLocation="CenterScreen">
9.     <Grid>
10.         <WebBrowser Name="web" />
11.     </Grid>
12.</Window>

```

- **Help.cs**

```

1. using System.Windows;
2.
3. namespace Activision_Mendeleyev_table
4. {
5.     ///<summary>
6.     /// Логика взаимодействия для Help.xaml
7.     ///</summary>
8.     public partial class Help : Window

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```
9.      {
10.          /// <summary>
11.          /// Конструктор справки, для упрощения использования программы
12.          /// </summary>
13.          public Help()
14.          {
15.              InitializeComponent();
16.              web.NavigateToString("<!DOCTYPE html>" +
17.                  "<html lang = 'en' xmlns = 'http://www.w3.org/1999/xhtml\''>" +
18.                  "<head>" +
19.                  "<meta http-equiv='Content-Type'
20.                  content='text/html; charset=UTF-8'>" +
21.                  "<title>Справка</title>" +
22.                  "</head>" +
23.                  "<body>" +
24.                  "<h1>Справка</h1>" +
25.                  "<ol style = 'list-style-type: upper-roman;'>" +
26.                  "<li><a href = '#Работа с элементом'> Работа с элементом </a>"
27.                  +
28.                  "<ol>" +
29.                  "<li> Заполнение таблицы </li>" +
30.                  "</ol>" +
31.                  "</li>" +
32.                  "<li><a href = '#Работа с соединением'> Работа с соединением
33.                  </a>" +
34.                  "<ol>" +
35.                  "<li> Выбор(создание) соединения </li>" +
36.                  "<li> Заполнение таблицы </li>" +
37.                  "</ol>" +
38.                  "</li>" +
39.                  "<li><a href = '#Работа с системой соединений'> Работа с
40.                  системой соединений </a>" +
41.                  "<ol>" +
42.                  "<li> Выбор(создание) системы соединений </li>" +
43.                  "<li> Заполнение таблицы </li>" +
44.                  "</ol>" +
45.                  "</li>" +
46.                  "<li><a href = '#Формат формул'> Формат формул </a></li>" +
47.                  "<li><a href = '#Описание функций'> Описание функций </a></li>"
48.                  +
49.                  "<li><a href = '#Работа с фазовыми диаграммами'> Работа с
50.                  фазовыми диаграммами </a></li>" +
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

45. "" +
46. "<h3 style = 'text-align: center;*> Работа с элементом </h3>" +
47. "<p align = 'center'> Заполнение таблицы </p>" +
48. "" +
49. " Для работы с элементом дважды кликните левой кнопкой мыши по соответствующему обозначению в Таблице Менделеева на стартовом окне;" +
50. " Для начала изменения таблицы необходимо нажать кнопку "Редактировать таблицу";" +
51. " Для добавления (удаления) строки и столбца необходимо нажать, соответственно, кнопки "Добавить строку"; ("Удалить строку";) и "Добавить столбец"; ("Удалить столбец";)" +
52. " Чтобы сохранить изменения, которые вы произвели в таблице, необходимо нажать кнопку "Сохранить"; (!ВАЖНО: заголовки таблицы сохраняются для всех элементов (удаление столбца может повлечь потерю данных));" +
53. " Заккрыть окно с таблицей можно нажав кнопку "x";." +
54. "" +
55. "<h3 style = 'text-align: center;*> Работа с соединением</h3>" +
56. "<p align = 'center'> Выбор (создание) соединения </p>" +
57. "" +
58. " Для работы с соединением нажмите кнопку "Работа с соединением"; в верхнем меню стартового окна;" +
59. " Если соединение уже создано, то вы можете выбрать его из выпадающего списка в данном окне или впишите его в текстовое поле и нажмите кнопку "Готово";" +
60. " Для создания нового соединения выпишите его в текстовое поле и нажмите кнопку "Готово";." +
61. "" +
62. "<p align = 'center'> Заполнение таблицы </p>" +
63. "" +
64. " Для начала изменения таблицы необходимо нажать кнопку "Редактировать таблицу";" +
65. " Для добавления (удаления) строки и столбца необходимо нажать, соответственно, кнопки "Добавить строку"; ("Удалить выделенные строки";) и "Добавить столбец"; ("Удалить столбец";)" +
66. " Для добавления формулы нужно нажать кнопку "Добавить формулу";" +

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

67. " Чтобы сохранить изменения, которые вы произвели в
таблице, и пересчитать формулы, необходимо нажать кнопку "Рассчитать и
сохранить";" +
68. " Заккрыть окно с таблицей можно нажав кнопку
"x";." +
69. " " +
70. " <h3 style = 'text-align: center;'><a name = 'Работа с системой
соединений'> Работа с системой соединений </h3>" +
71. " <p align = 'center'> Выбор (создание) соединения </p>" +
72. " " +
73. " Для работы с системой соединений нажмите кнопку
"Работа с системой соединений"; в верхнем меню стартового
окна;" +
74. " Если система уже создана, то вы можете выбрать её из
выпадающего списка в данном окне или впишите её в текстовое поле и нажмите
кнопку "Готово";" +
75. " Для создания новой системы соединения выпишите её в
текстовое поле и нажмите кнопку "Готово";." +
76. " " +
77. " <p align = 'center'> Заполнение таблицы </p>" +
78. " " +
79. " Для начала изменения таблицы необходимо нажать кнопку
"Редактировать таблицу"; " +
80. " Для добавления (удаления) строки и столбца необходимо
нажать, соответственно, кнопки "Добавить строку"; ("Удалить
выделенные строки";) и"Добавить столбец"; ("Удалить
столбец";)" +
81. " Для добавления формулы нужно нажать кнопку "Добавить
формулу"; " +
82. " Столбец "X"; нельзя удалить, и можно заполнять
только вещественными числами, в виде десятичных дробей;" +
83. " Чтобы сохранить изменения, которые вы произвели в
таблице, и пересчитать формулы, необходимо нажать кнопку "Рассчитать и
сохранить";" +
84. " Заккрыть окно с таблицей можно нажав кнопку
"x";." +
85. " " +
86. " <h3 style = 'text-align: center;'>
Формат формул </h3>" +
87. " <p style='margin-left: 20px'>В формулу могут входить:</p>" +
88. " <ol style = 'list-style-type: lower-alpha;'>" +
89. " Вещественные числа, в виде десятичных дробей;" +

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

90. " Конструкции, по которым ищется значение в таблицах:" +
91. "<ol style = 'list-style-type: disc;'" +
92. " <обозначение искомого
свойства>{<элемент/соединение/система>} - получает значение искомого свойства
элемента/соединения/системы соединений находящегося в первой строке таблицы
свойств;" +
93. " <обозначение искомого свойства>{<вспомогательное
свойство, по которому ищем искомое>{<значение вспомогательного
свойства>}<элемент/соединение/система>} - получает значение искомого свойства
элемента/соединения/системы соединений соответствующего значению
вспомогательного свойства из таблицы свойств);" +
94. " <элемент/соединение/система>{<номер строки>;<номер
столбца>} - получает значение ячейки таблицы свойств
элемента/соединения/системы соединений на пересечении заданной строки и
столбца (нумерация строк и столбцов начинается с нуля);" +
95. " <элемент/соединение/система>{<номер строки>;<обозначение
искомого свойства>} - получает значение искомого свойства
элемента/соединения/системы соединений на заданной строке (нумерация строк
начинается с нуля);" +
96. "" +
97. " Символы математических операций("+";, "-
";, "*";, "/";, "%"; - остаток от
деления);" +
98. " Скобки, обозначающие приоритет математических
действий("("; и ")";);" +
99. " Функции, описанные ниже (если аргументов больше одного -
следует их разделять символом ";";. Пример: min(32; sin(4))." +
100. "<p>Необходимо наличие элементов/соединений/систем
соединений и свойств им принадлежащих, которые используются в формуле, в
таблицах.</p>" +
101. "" +
102. "<h3 style = 'text-align: center;'"<a name = 'Описание
функций'"> Описание функций </h3>" +
103. "" +
104. " sin(a) - вычисляет синус a;" +
105. " cos(a) - вычисляет косинус a;" +
106. " tan(a) - вычисляет тангенс a;" +
107. " ctan(a) - вычисляет котангенс a;" +
108. " abs(a) - вычисляет модуль a;" +
109. " ln(a) - вычисляет натуральный логарифм a;" +
110. " exp(a) - вычисляет экспоненту степени a;" +
111. " min(a; b) - находит минимум из a и b;" +

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

112.      "<li> max(a; b) - находит максимум из a и b;</li>" +
113.      "<li> pow(a; b) - возводит a в степень b.</li>" +
114.      "</ol>" +
115.      "<h3 style = 'text-align: center;'><a name = 'Работа с  

    фазовыми диаграммами'> Работа с фазовыми диаграммами </a></h3>" +
116.      "<p align = 'center'><b> Купол распада </b></p>" +
117.      "<ol>" +
118.      "<li> Для начала построения купола распада необходимо  

    нажать кнопку &quot;Купол распада&quot;; в окне редактирования таблицы  

    соответствующей системы соединений;</li>" +
119.      "<li> Далее в открывшемся окне необходимо задать  

    некоторые параметры системы необходимые для построения купола(это же окно можно  

    вызвать нажав на пункт меню &quot;Настройка данных&quot;);</li>" +
120.      "<li> Для построения купола распада необходимо нажать  

    кнопку &quot;Построить купол&quot;;</li>" +
121.      "<li> Для настройки предельных значений температуры(оси  

    Y) необходимо выбрать пункт меню &quot;Графическое меню&quot;;->&quot;Настройка  

    границ&quot;; и ввести значения верхних и нижних границ температуры;</li>" +
122.      "<li> Добавление экспериментальных данных происходит  

    путем ввода точек в таблицу в правой части окна или загрузки из файла при  

    нажатии на кнопку&quot;Загрузить точки&quot;;(чтобы сохранить точки из таблицы в  

    файл необходимо нажать кнопку &quot;Сохранить точки&quot;);</li>" +
123.      "<li> Удалить точки можно выделив их в таблице и нажав  

    после клавишу DELETE.</li>" +
124.      "<li> Для изменения формата отображения экспериментальных  

    данных необходимо поставить/снять отметку пункта меню &quot;Графическое  

    меню&quot;;->&quot;Эксперимент точками&quot;;</li>" +
125.      "<li> Также в пункте меню &quot;Графическое меню&quot;;-  

    >&quot;Настройка цвета&quot;; можно выбрать цвет каждого графика.</li>" +
126.      "</ol>" +
127.      "<p align = 'center'><b> Аппроксимация функции </b></p>"
    +
128.      "<ol>" +
129.      "<li> Для аппроксимации термодинамической функции  

    смещения необходимо нажать кнопку &quot;Построить функцию Hsm и  

    аппроксимировать&quot;; в окне построения купола распада;</li>" +
130.      "<li> Если не были заданы точки, то аппроксимация не  

    будет проведена;</li>" +
131.      "<li> Для настройки предельных значений температуры(оси  

    Y) необходимо выбрать пункт меню &quot;Графическое меню&quot;;->&quot;Настройка  

    границ&quot;; и ввести значения верхних и нижних границ температуры;</li>" +

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

132.             "<li> После аппроксимации, если нажать на кнопку
                &quot;Построить купол&quot;;, будет предложено использовать новые значения
                параметров для его построения.</li>" +
133.             "</ol>" +
134.             "<p align = 'center'><b> Оценка чувствительности
                </b></p>" +
135.             "<ol>" +
136.             "<li> Для начала работы с оценкой чувствительности
                термодинамической функции смещения необходимо нажать пункт меню &quot;Оценка
                чувствительности&quot;; в окне построения купола распада;</li>" +
137.             "<li> При изменении слайдеров в правой половине окна
                будет меняться функция в зависимости от выбранных значений (текущие значения
                отображаются над слайдерами);</li>" +
138.             "<li> Для настройки предельных значений слайдеров
                необходимо выбрать пункт меню &quot;Графическое меню&quot;;->&quot;Настройка
                границ&quot;; и ввести значения верхних и нижних границ параметров;</li>" +
139.             "<li> Для восстановления начальных значений параметров
                необходимо снова нажать пункт меню &quot;Оценка чувствительности&quot;;.</li>" +
140.             "</ol>" +
141.             "</body>" +
142.             "</html>");
143.         }
144.     }
145. }
146.

```

11. Классы Approximation:

- Criterion.cs

```

1. using Activision_Mendeleyev_table.HelperClasses;
2. using System;
3. using System.Collections.Generic;
4.
5. namespace Activision_Mendeleyev_table.Approximation
6. {
7.     /// <summary>
8.     /// Класс, содержащий различные критерии оценки отклонения функции
9.     /// </summary>
10.    public static class Criterion
11.    {
12.        /// <summary>
13.        /// Критерий  $\max|f-y|$  оценки отклонения  $F(x)$  от точек

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

14.      /// </summary>
15.      /// <param name="tab">лист точек</param>
16.      /// <param name="F">функция</param>
17.      /// <param name="par">начальное значение параметров функции</param>
18.      /// <returns>значение отклонения</returns>
19.      public static double Criterion_MAX(List<Point> tab, Func<double,
double[], double> F, double[] par)
20.      {
21.          double max = 0, f, s = 0;
22.          foreach (Point mp in tab)
23.          {
24.              f = F(mp.X, par);
25.              s = Math.Abs(f - mp.Y);
26.              max = max > s ? max : s;
27.          }
28.
29.          return max;
30.      }
31.
32.      /// <summary>
33.      /// Критерий  $(\sum |f-y|^2)/N$  оценки отклонения  $F(x)$  от точек
34.      /// </summary>
35.      /// <param name="tab">лист точек</param>
36.      /// <param name="F">функция</param>
37.      /// <param name="par">начальное значение параметров функции</param>
38.      /// <returns>значение отклонения</returns>
39.      public static double Criterion_CKO(List<Point> tab, Func<double,
double[], double> F, double[] par)
40.      {
41.          double f, sum = 0;
42.          foreach (Point mp in tab)
43.          {
44.              f = F(mp.X, par);
45.              sum += Math.Pow((f - mp.Y), 2);
46.          }
47.
48.          return Math.Sqrt(sum / tab.Count);
49.      }
50.  }
51. }
52.

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

- **Library.cs**

```

1. using Activision_Mendeleyev_table.HelperClasses;
2. using System;
3. using System.Collections.Generic;
4.
5. namespace Activision_Mendeleyev_table.Approximation
6. {
7.     /// <summary>
8.     /// Класс, содержащий методы аппроксимации функции
9.     /// </summary>
10.    public class Library
11.    {
12.        /// <summary>
13.        /// Метод аппроксимации набора точек заданной функцией (Function)
14.        /// </summary>
15.        /// <param name="tab">лист точек</param>
16.        /// <param name="Function">аппроксимирующая функция</param>
17.        /// <param name="Par">начальное значение параметров функции</param>
18.        /// <param name="ApproxAccuracy">метод оценки точности
19.        /// аппроксимации</param>
20.        /// <returns>новый набор параметров функции</returns>
21.        public static double[] AproxiTab(List<Point> tab, Func<double,
22.        double[], double> Function, double[] Par,
23.        Func<List<Point>, Func<double, double[], double>, double[], double>
24.        ApproxAccuracy)
25.        {
26.            // Локальный метод
27.            double funN(double[] par)
28.            {
29.                if (tab.Count == 0)
30.                    throw new ArgumentNullException("", new
31.                    Exception("MyException"));
32.                double result = ApproxAccuracy(tab, Function, par);
33.                return result;
34.            }
35.            double[] res = GradientMinimization(funN, Par, 1E-8, 1E-11, 10000);
36.            return res;
37.        }
38.    }
39. }

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

37.      /// <summary>
38.      /// Вычисление градиента и направляющих вектора перемещения
39.      /// </summary>
40.      /// <param name="funN">исследуемая функция</param>
41.      /// <param name="X0">вектор параметров - исследуемая точка</param>
42.      /// <param name="del">относительная ??вариация??? каждого
    параметра</param>
43.      /// <returns>вектор перемещения вдоль градиента</returns>
44.      public static double[] Gradient(Func<double[], double> funN, double[]
X0, double del = 0.001)
45.      {
46.          //double del=0.00001)
47.          //double del = 0.01)
48.          {
49.              int NC = X0.Length;
50.              double[] G = new double[NC];
51.              double[] dx = new double[NC];
52.              for (int j = 0; j < NC; j++)
53.                  if (X0[j] == 0)
54.                      dx[j] = del;
55.                  else
56.                      dx[j] = Math.Abs(X0[j] * del);
57.
58.              double[] V = new double[NC];
59.              X0.CopyTo(V, 0);
60.              double Fma, Fmi, dFi;
61.              for (int j = 0; j < NC; j++)
62.              {
63.                  V[j] = X0[j] + dx[j];
64.                  Fma = funN(V);
65.                  V[j] = X0[j] - dx[j];
66.                  Fmi = funN(V);
67.                  dFi = Fma - Fmi;
68.                  G[j] = dFi / (2 * dx[j]);
69.                  V[j] = X0[j];
70.              }
71.
72.              // unit vector along gradient:
73.              double[] S = new double[NC];
74.              double len = 0;
75.              for (int j = 0; j < NC; j++)
76.                  len += Math.Pow(G[j], 2);

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

77.         len = Math.Sqrt(len);
78.         for (int j = 0; j < NC; j++)
79.             S[j] = -G[j] / len;
80.
81.         return S;
82.     }
83.
84.     /// <summary>
85.     /// Вычисление минимума вдоль градиента
86.     /// </summary>
87.     /// <param name="funN">исследуемая функция</param>
88.     /// <param name="X0">начальный вектор параметров</param>
89.     /// <param name="Delta">допустимое отклонение для ширины
    интервала</param>
90.     /// <param name="Epsilon">допустимое отклонение для  $|f(b) -$ 
     $f(a)|$ </param>
91.     /// <returns>параметры (точка) минимума вдоль градиента</returns>
92.     private static double[] Quadmin(Func<double[], double> funN, double[]
    X0, double Delta = 1E-5, double Epsilon = 1E-7)
93.     {
94.         int NC = X0.Length;
95.         double Y0 = funN(X0);
96.         double[] P0 = new double[NC];
97.         X0.CopyTo(P0, 0);
98.         double[] P1 = new double[NC];
99.         double[] P2 = new double[NC];
100.        double[] S = new double[NC];
101.        double H = 1.0;
102.        double Err = 1.0;
103.        int Jmax = 20;
104.        double H0, H1, H2, Hmin, E0, E1, E2, Y1, Y2, D, Ymin;
105.        int i;
106.        int Cond = 0;
107.        int J = 0;
108.
109.        S = Gradient(funN, X0);
110.        for (i = 0; i < NC; i++)
111.        {
112.            P1[i] = P0[i] + H * S[i];
113.            P2[i] = P0[i] + 2.0 * H * S[i];
114.        }
115.

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

91
RU.17701729.04.15-01 12 01-1

```

116.      Y1 = funN(P1);
117.      Y2 = funN(P2);
118.      double[] Pmin = new double[NC];
119.      while ((J < Jmax) & (Cond == 0))
120.      {
121.          if (Y0 < Y1) /* Make H smaller */
122.          {
123.              Y2 = Y1;
124.              H = H / 2.0;
125.              for (i = 0; i < NC; i++)
126.              {
127.                  P2[i] = P1[i];
128.                  P1[i] = P0[i] + H * S[i];
129.              }
130.              Y1 = funN(P1);
131.          }
132.          else
133.          {
134.              if (Y2 < Y1) /* Make H larger */
135.              {
136.                  Y1 = Y2;
137.                  H = 2.0 * H;
138.                  for (i = 0; i < NC; i++)
139.                  {
140.                      P1[i] = P2[i];
141.                      P2[i] = P0[i] + 2.0 * H * S[i];
142.                  }
143.                  Y2 = funN(P2);
144.              }
145.              else Cond = -1;
146.          }
147.      }
148.
149.      if (H < Delta) Cond = 1;
150.      D = 4.0 * Y1 - 2.0 * Y0 - 2.0 * Y2;
151.      /* Quadratic interpolation to find Hmin */
152.      if (D < 0)
153.          Hmin = H * (4.0 * Y1 - 3.0 * Y0 - Y2) / D;
154.      else /* check division by zero */
155.      {
156.          Cond = 4;
157.          Hmin = H / 3.0;

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

158.         }
159.
160.         for (i = 0; i < NC; i++)
161.             Pmin[i] = P0[i] + Hmin * S[i];
162.
163.         Ymin = funN(Pmin);
164.
165.         /* Convergence test for the points */
166.         H0 = Math.Abs(Hmin);
167.         H1 = Math.Abs(Hmin - H);
168.         H2 = Math.Abs(Hmin - 2.0 * H);
169.         if (H0 < H) H = H0;
170.         if (H1 < H) H = H1;
171.         if (H2 < H) H = H2;
172.         if (H < Delta) Cond = 1;
173.
174.         /* Convergence test for the function values */
175.         E0 = Math.Abs(Y0 - Ymin);
176.         E1 = Math.Abs(Y1 - Ymin);
177.         E2 = Math.Abs(Y2 - Ymin);
178.         if (E0 < Err)
179.             Err = E0;
180.         else if (E1 < Err)
181.             Err = E1;
182.         else if (E2 < Err)
183.             Err = E2;
184.         else if ((E0 == 0) && (E1 == 0) && (E2 == 0))
185.             Err = 0;
186.
187.         if (Err < Epsilon)
188.             Cond = 2;
189.         if ((Cond == 2) && (H < Delta))
190.             Cond = 3;
191.
192.         J++;
193.         return Pmin;
194.     }
195.
196.     /// <summary>
197.     /// Метод наискорейшего спуска (метод градиентной минимизации)
198.     /// </summary>
199.     /// <param name="funN">исследуемая функция</param>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

200.          /// <param name="X0">начальный вектор параметров</param>
201.          /// <param name="Delta">допустимое отклонение для ширины
            интервала</param>
202.          /// <param name="Epsilon">допустимое отклонение для  $|f(b) -$ 
             $f(a)|$ </param>
203.          /// <param name="Max">максимальное число итераций</param>
204.          /// <returns>оптимальный вектор параметров</returns>
205.          public static double[] GradientMinimization(Func<double[],
            double> funN, double[] X0, double Delta = 1E-8, double Epsilon = 1E-11, int Max
            = 100)
206.          {
207.              int NC = X0.Length;
208.              double[] Q1 = new double[NC];
209.              double[] Q2 = new double[NC];
210.              double F1, F2;
211.              int iter = 0;
212.              double deltaX = 0;
213.              X0.CopyTo(Q2, 0);
214.
215.              do
216.              {
217.                  iter++;
218.                  Q2.CopyTo(Q1, 0);
219.                  F1 = funN(Q1);
220.                  Q2 = Quadmin(funN, Q1);
221.                  F2 = funN(Q2);
222.                  deltaX = 0;
223.                  for (int k = 0; k < NC; k++)
224.                  {
225.                      double ZN = Math.Abs(Q2[k]);
226.                      if (ZN > 0.0)
227.                      {
228.                          double DR = Math.Abs((Q2[k] - Q1[k]) / ZN);
229.                          deltaX = DR > deltaX ? DR : deltaX;
230.                      }
231.                  }
232.              }
233.              while (iter < Max & Math.Abs(F1 - F2) > Epsilon & deltaX >
                Delta);
234.              double deltaF = (F1 - F2);
235.              return Q2;
236.          }

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

237. }
238. }
239.

12. Классы DrawingClasses:

- Collapse.cs

```
1. using Activision_Mendeleyev_table.HelperClasses;  
2. using System;  
3. using System.Linq;  
4. using System.Xml.Linq;  
5.  
6. namespace Activision_Mendeleyev_table.DrawingClasses  
7. {  
8.     /// <summary>  
9.     /// Класс, содержащий набор точек фазовой диаграммы и методы их получения  
10.    /// </summary>  
11.    class Collapse  
12.    {  
13.        /// <summary>  
14.        /// Набор точек правого соединения  
15.        /// </summary>  
16.        public Point[] right;  
17.  
18.        /// <summary>  
19.        /// Набор точек левого соединения  
20.        /// </summary>  
21.        public Point[] left;  
22.  
23.        /// <summary>  
24.        /// Получает точки для фазовой диаграммы  
25.        /// </summary>  
26.        /// <param name="system">система соединений</param>  
27.        public Collapse(BinSystem system)  
28.        {  
29.            string r = GetRatio(system.delR / Math.Min(system.R1, system.R2));  
30.  
31.            System.Windows.Resources.StreamResourceInfo ri =  
                System.Windows.Application.GetResourceStream(new  
                    Uri("DrawingClasses/Collapse.xml", UriKind.Relative));  
32.            System.IO.Stream data = ri.Stream;  
33.
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

34.         XDocument doc = XDocument.Load(data);
35.         string[] x1values = doc.Root.Elements().First(
36.             p => p.Attribute("ratio").Value ==
37.             r).Element("x1").Value.Split(';');
38.         string[] x2values = doc.Root.Elements().First(
39.             p => p.Attribute("ratio").Value ==
40.             r).Element("x2").Value.Split(';');
41.         string[] y1values = doc.Root.Elements().First(
42.             p => p.Attribute("ratio").Value ==
43.             r).Element("y1").Value.Split(';');
44.         string[] y2values = doc.Root.Elements().First(
45.             p => p.Attribute("ratio").Value ==
46.             r).Element("y2").Value.Split(';');
47.
48.         right = new Point[x1values.Length];
49.         left = new Point[x2values.Length];
50.
51.         for (int i = 0; i < x1values.Length; i++)
52.             right[i] = new Point(double.Parse(x1values[i]),
53.                 double.Parse(y1values[i]));
54.
55.         for (int i = 0; i < x2values.Length; i++)
56.             left[i] = new Point(double.Parse(x2values[i]),
57.                 double.Parse(y2values[i]));
58.     }
59.
60.     /// <summary>
61.     /// Получает соотношение радиусов
62.     /// </summary>
63.     /// <param name="ratio">delR/Rmin</param>
64.     /// <returns>соотношение радиусов</returns>
65.     private string GetRatio(double ratio)
66.     {
67.         ratio = Math.Round(ratio, 3);
68.         if ((ratio >= 0) && (ratio < 0.025)) return "0,00";
69.         else if ((ratio <= 0.05) || ((ratio > 0.05) && (ratio < 0.075)))
70.             return "0,05";
71.         else if ((ratio <= 0.1) || ((ratio > 0.1) && (ratio < 0.125)))
72.             return "0,10";
73.         else if ((ratio <= 0.15) || ((ratio > 0.15) && (ratio < 0.175)))
74.             return "0,15";
75.     }

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата


```

66.         else if ((ratio <= 0.20) || ((ratio > 0.20) && (ratio < 0.225)))
            return "0,20";
67.         else if ((ratio <= 0.25) || ((ratio > 0.25) && (ratio < 0.275)))
            return "0,25";
68.         else if ((ratio <= 0.30) || ((ratio > 0.30) && (ratio < 0.325)))
            return "0,30";
69.         else throw new Exception("Недопустимое отношение радиусов!", new
            Exception("MyException"));
70.     }
71. }
72. }
73.

```

- **CollapseGraph.cs**

```

1. using Activision_Mendelejev_table.HelperClasses;
2. using System;
3. using System.Collections.Generic;
4. using System.Drawing;
5. using Point = System.Drawing.Point;
6.
7. namespace Activision_Mendelejev_table.DrawingClasses
8. {
9.     /// <summary>
10.    /// Класс, отрисовывающий фазовую диаграмму
11.    /// </summary>
12.    class CollapseGraph
13.    {
14.        /// <summary>
15.        /// Карандаш теоритического соотношения
16.        /// </summary>
17.        private static Pen pen = Pens.Black;
18.        /// <summary>
19.        /// Карандаш экспериментального соотношения
20.        /// </summary>
21.        private static Pen penExp = Pens.Red;
22.        /// <summary>
23.        /// Карандаш аппроксимированного соотношения
24.        /// </summary>
25.        private static Pen penApp = Pens.Green;
26.        /// <summary>
27.        /// Точки эксперимента

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

28.      /// </summary>
29.      private static List<PointF> experiment = new List<PointF>();
30.      /// <summary>
31.      /// Поверхность для рисования
32.      /// </summary>
33.      private Graphics g;
34.      /// <summary>
35.      /// Система соединений
36.      /// </summary>
37.      private BinSystem system;
38.      /// <summary>
39.      /// Ширина поля для диаграммы
40.      /// </summary>
41.      private readonly int width;
42.      /// <summary>
43.      /// Точки правого соединения (теория)
44.      /// </summary>
45.      private Point[] right;
46.      /// <summary>
47.      /// Точки левого соединения (теория)
48.      /// </summary>
49.      private Point[] left;
50.
51.      /// <summary>
52.      /// Конструктор
53.      /// </summary>
54.      /// <param name="g">поверхность для рисования</param>
55.      /// <param name="system">система соединений</param>
56.      /// <param name="width">ширина поля для диаграммы</param>
57.      public CollapseGraph(Graphics g, BinSystem system, int width)
58.      {
59.          this.width = width - 80;
60.          this.g = g;
61.          this.system = system;
62.      }
63.
64.      /// <summary>
65.      /// Свойство цвета теоритического соотношения
66.      /// </summary>
67.      public static Color Color
68.      {
69.          get { return pen.Color; }

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

70.         set { pen = new Pen(value); }
71.     }
72.
73.     /// <summary>
74.     /// Свойство цвета экспериментального соотношения
75.     /// </summary>
76.     public static Color ExperimentColor
77.     {
78.         get { return penExp.Color; }
79.         set { penExp = new Pen(value); }
80.     }
81.
82.     /// <summary>
83.     /// Свойство цвета аппроксимированного соотношения
84.     /// </summary>
85.     public static Color ApproximationColor
86.     {
87.         get { return penApp.Color; }
88.         set { penApp = new Pen(value); }
89.     }
90.
91.     /// <summary>
92.     /// Рисует купол распада
93.     /// </summary>
94.     public void DrawCollapse()
95.     {
96.         Collapse collapse = new Collapse(system);
97.         UpTemp = UpTemp == -1 ? (int)system.Tmax : UpTemp;
98.         DownTemp = DownTemp == -1 ? (int)(0.20 * system.Tmax) : DownTemp;
99.
100.         right = new Point[collapse.right.Length];
101.         for (int i = 0; i < right.Length; i++)
102.         {
103.             int x = 80 + (int)(width * (1 - collapse.right[i].X));
104.             int y = width - 40 - (int)(width * ((collapse.right[i].Y
105. * system.Tmax - DownTemp) / (UpTemp - DownTemp)));
106.             y = y > width ? width : y;
107.
108.             right[i] = new Point(x, y);
109.         }
110.
111.         left = new Point[collapse.left.Length];

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

111.         for (int i = 0; i < left.Length; i++)
112.         {
113.             int x = 80 + (int)(width * collapse.left[i].X);
114.             int y = width - 40 - (int)(width * ((collapse.left[i].Y *
system.Tmax - DownTemp) / (UpTemp - DownTemp)));
115.             y = y > width ? width : y;
116.
117.             left[i] = new Point(x, y);
118.         }
119.
120.         g.DrawString("T, °C", new Font("X", 14), Brushes.Black, new
Point(80, 0));
121.         g.DrawLine(pen, right);
122.         g.DrawLine(pen, left);
123.     }
124.
125.     /// <summary>
126.     /// Рисует термодинамическую функцию смещения
127.     /// </summary>
128.     /// <param name="f">флаг: true - аппроксимация, false -
теория</param>
129.     public void DrawDH(bool f = true)
130.     {
131.         DownTemp = DownTemp == -1 ? 0 : DownTemp;
132.         UpTemp = UpTemp == -1 ? (int)(system.Hsm(0.5) * 1000) + 100 :
UpTemp;
133.
134.         Point[] dh = new Point[21];
135.         for (double i = 0; i < 1; i += 0.05)
136.             dh[(int)Math.Round(i * 20)] = new Point(80 + (int)(i *
width), width - 40 - (int)((system.Hsm(i) * 1000 - DownTemp) / (UpTemp -
DownTemp) * width));
137.
138.         dh[20] = new Point(80 + width, width - 40);
139.         g.DrawString("ΔHсм, ккал/моль", new Font("X", 14),
Brushes.Black, new Point(80, 0));
140.         if (f)
141.             g.DrawLine(penApp, dh);
142.         else
143.             g.DrawLine(pen, dh);
144.     }
145.

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

146.          /// <summary>
147.          /// Рисует оси координат
148.          /// </summary>
149.          public void DrawAxes()
150.          {
151.              g.DrawString(system.ElementA + system.ElementX, new Font("X",
152.                  14), Brushes.Black, new Point(80, width + 30));
153.              g.DrawString(system.ElementB + system.ElementX, new Font("X",
154.                  14), Brushes.Black, new Point(width - 100, width + 30));
155.              g.DrawLine(Pens.Black, 80, 0, 80, width + 30);
156.              g.DrawLine(Pens.Black, 30, width - 40, width + 80, width -
157.                  40);
158.              for (double x = 0; x <= 1; x += 0.1)
159.              {
160.                  g.DrawString(x <= 0.5 ? x.ToString() : (1 -
161.                      x).ToString(), new Font("X", 12), Brushes.Black,
162.                          80 + (float)(width * x), width - 30);
163.                  g.DrawLine(Pens.Black, 80 + (int)(width * x), width - 45,
164.                      80 + (int)(width * x), width - 35);
165.              }
166.              g.DrawString("0", new Font("X", 12), Brushes.Black, width +
167.                  30, width - 30);
168.              double c = Math.Round((UpTemp - DownTemp) / 100.0);
169.              if (c < (UpTemp - DownTemp) / 100)
170.                  c = (c + 1) * 10;
171.              else
172.                  c *= 10;
173.              for (double x = DownTemp; x <= UpTemp; x += c)
174.              {
175.                  g.DrawString(x.ToString(), new Font("X", 12),
176.                      Brushes.Black, 0, width - 40 - (int)((x - DownTemp) / (UpTemp - DownTemp) *
177.                          width));
178.                  g.DrawLine(Pens.Black, 75, width - 40 - (int)((x -
179.                      DownTemp) / (UpTemp - DownTemp) * width), 85, width - 40 - (int)((x - DownTemp)
180.                          / (UpTemp - DownTemp) * width));
181.              }
182.          }
183.      }

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

178.      /// <summary>
179.      /// Рисует эксперимент
180.      /// </summary>
181.      public void DrawExperiment()
182.      {
183.          List<Point> left = new List<Point>();
184.          List<Point> right = new List<Point>();
185.
186.          foreach (var item in experiment)
187.          {
188.
189.              int x = 80 + (int)(item.X * width);
190.              int y = width - 40 - (int)((item.Y - DownTemp) / (UpTemp
- DownTemp) * width);
191.
192.              y = y > width ? width : y;
193.
194.              if (item.X <= 0.5)
195.                  left.Add(new Point(x, y));
196.              else
197.                  right.Add(new Point(x, y));
198.          }
199.
200.          if (!ExperimentIsPoints)
201.          {
202.              Point[] arrLeft = left.ToArray();
203.              Point[] arrRight = right.ToArray();
204.
205.              Array.Sort(arrLeft, (x, y) => x.X.CompareTo(y.X));
206.              Array.Sort(arrRight, (x, y) => x.X.CompareTo(y.X));
207.
208.              if (left.Count > 1)
209.                  g.DrawLine(penExp, arrLeft);
210.
211.              if (right.Count > 1)
212.                  g.DrawLine(penExp, arrRight);
213.          }
214.          else
215.          {
216.              foreach (var item in left)
217.              {

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

102
RU.17701729.04.15-01 12 01-1

```

218.             g.FillEllipse(penExp.Brush, item.X - 6, item.Y - 6,
12, 12);
219.             }
220.         foreach (var item in right)
221.         {
222.             g.FillEllipse(penExp.Brush, item.X - 6, item.Y - 6,
12, 12);
223.         }
224.     }
225. }
226.
227.     /// <summary>
228.     /// Свойство, определяющее вид отрисовки эксперимента
229.     /// </summary>
230.     public static bool ExperimentIsPoints { get; set; } = true;
231.
232.     /// <summary>
233.     /// Свойство верхней границы температуры(графика по Y)
234.     /// </summary>
235.     public static int UpTemp { get; set; } = -1;
236.
237.     /// <summary>
238.     /// Свойство нижней границы температуры(графика по Y)
239.     /// </summary>
240.     public static int DownTemp { get; set; } = -1;
241.
242.     /// <summary>
243.     /// Добавляет точку в эксперимент
244.     /// </summary>
245.     /// <param name="x1">координата X</param>
246.     /// <param name="t">координата Y</param>
247.     public static void AddExperimentalPoint(double x1, double t)
248.     {
249.         experiment.Add(new PointF((float)x1, (float)t));
250.     }
251.
252.     /// <summary>
253.     /// Удаляет выбранную точку из эксперимента
254.     /// </summary>
255.     public static void RemoveSelectedPoint(int i)
256.     {
257.         if (experiment.Count > i)

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```
258.             experiment.RemoveAt(i);
259.         }
260.
261.         /// <summary>
262.         /// Удаляет все точки из эксперимента
263.         /// </summary>
264.         public static void ClearExperiment()
265.         {
266.             experiment.Clear();
267.         }
268.     }
269. }
270.
```

13. Классы HelperClasses:

- BinSystem.cs

```
1. using System;
2.
3. namespace Activision_Mendeleyev_table.HelperClasses
4. {
5.     /// <summary>
6.     /// Класс, представляющий бинарную систему соединений
7.     /// </summary>
8.     public class BinSystem
9.     {
10.        /// <summary>
11.        /// Массив обозначений в таблицах данных
12.        /// </summary>
13.        public string[] symbols = new string[3] { "R(i)", "x", "ФЗ" };
14.        /// <summary>
15.        /// Универсальная газовая постоянная
16.        /// </summary>
17.        private const double kN = 1.9844 * 0.001;
18.        /// <summary>
19.        /// Константа Моделунга
20.        /// </summary>
21.        public double A = -1;
22.        /// <summary>
23.        /// Формальный заряд общего химического элемента
24.        /// </summary>
25.        public double zX = -1;
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата


```

26.      /// <summary>
27.      /// Номер строки из таблицы данных первого химического элемента
28.      /// </summary>
29.      public int numA = 0;
30.      /// <summary>
31.      /// Номер строки из таблицы данных второго химического элемента
32.      /// </summary>
33.      public int numB = 0;
34.      /// <summary>
35.      /// Номер строки из таблицы данных общего химического элемента
36.      /// </summary>
37.      public int numX = 0;
38.      /// <summary>
39.      /// Обозначение системы соединений
40.      /// </summary>
41.      private readonly string sourceString;
42.      /// <summary>
43.      /// Первый химический элемент
44.      /// </summary>
45.      private Composition elemA;
46.      /// <summary>
47.      /// Второй химический элемент
48.      /// </summary>
49.      private Composition elemB;
50.      /// <summary>
51.      /// Общий химический элемент
52.      /// </summary>
53.      private Composition elemX;
54.
55.      /// <summary>
56.      /// Эмпирический параметр
57.      /// </summary>
58.      private double c = -1;
59.      /// <summary>
60.      /// Число структурных единиц
61.      /// </summary>
62.      private double m = -1;
63.      /// <summary>
64.      /// Координационное число
65.      /// </summary>
66.      private double n = -1;
67.      /// <summary>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

68.      /// Формальный заряд
69.      /// </summary>
70.      private double z = -1;
71.      /// <summary>
72.      /// Ионный радиус первого элемента
73.      /// </summary>
74.      private double r_1 = -1;
75.      /// <summary>
76.      /// Ионный радиус второго элемента
77.      /// </summary>
78.      private double r_2 = -1;
79.      /// <summary>
80.      /// Ионный радиус общего элемента
81.      /// </summary>
82.      private double r_3 = -1;
83.      /// <summary>
84.      /// Межатомное расстояние
85.      /// </summary>
86.      public double R_const = -1;
87.      /// <summary>
88.      /// Разность степеней ионности
89.      /// </summary>
90.      private double deleps = -1;
91.
92.      /// <summary>
93.      /// Возвращает ионный радиус первого элемента
94.      /// </summary>
95.      public double r1
96.      {
97.          get
98.          {
99.              if (r_1 == -1)
100.                  double.TryParse(elemA.Properties.Find(x =>
101.                      x.First.Second == symbols[0]).Second[numA], out r_1);
102.              return r_1;
103.          }
104.      }
105.
106.      /// <summary>
107.      /// Возвращает ионный радиус второго элемента
108.      /// </summary>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

109.         public double r2
110.     {
111.         get
112.         {
113.             if (r_2 == -1)
114.                 double.TryParse(elemB.Properties.Find(x =>
x.First.Second == symbols[0]).Second[numB], out r_2);
115.
116.             return r_2;
117.         }
118.     }
119.
120.     /// <summary>
121.     /// Возвращает ионный радиус общего элемента
122.     /// </summary>
123.     public double r3
124.     {
125.         get
126.         {
127.             if (r_3 == -1)
128.                 double.TryParse(elemX.Properties.Find(x =>
x.First.Second == symbols[0]).Second[numX], out r_3);
129.
130.             return r_3;
131.         }
132.     }
133.
134.     /// <summary>
135.     /// Межатомное расстояние первого соединения
136.     /// </summary>
137.     public double R1
138.     {
139.         get
140.         {
141.             if (r_1 == -1)
142.                 double.TryParse(elemA.Properties.Find(x =>
x.First.Second == symbols[0]).Second[numA], out r_1);
143.             if (r_3 == -1)
144.                 double.TryParse(elemX.Properties.Find(x =>
x.First.Second == symbols[0]).Second[numX], out r_3);
145.
146.             if (r_3 != -1 && r_1 != -1)

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

147.         return r_1 + r_3;
148.     else
149.         return -1;
150.     }
151. }
152.
153.     /// <summary>
154.     /// Межатомное расстояние второго соединения
155.     /// </summary>
156.     public double R2
157.     {
158.         get
159.         {
160.             if (r_2 == -1)
161.                 double.TryParse(elemB.Properties.Find(x =>
162. x.First.Second == symbols[0]).Second[numB], out r_2);
162.             if (r_3 == -1)
163.                 double.TryParse(elemX.Properties.Find(x =>
164. x.First.Second == symbols[0]).Second[numX], out r_3);
164.
165.             if (r_3 != -1 && r_2 != -1)
166.                 return r_2 + r_3;
167.             else
168.                 return -1;
169.         }
170.     }
171.
172.     /// <summary>
173.     /// Обозначение первого химического элемента
174.     /// </summary>
175.     public string ElementA
176.     {
177.         get { return elemA.Name; }
178.     }
179.
180.     /// <summary>
181.     /// Обозначение второго химического элемента
182.     /// </summary>
183.     public string ElementB
184.     {
185.         get { return elemB.Name; }
186.     }

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

187.
188.      /// <summary>
189.      /// Обозначение общего химического элемента
190.      /// </summary>
191.      public string ElementX
192.      {
193.          get { return elemX.Name; }
194.      }
195.
196.      /// <summary>
197.      /// Конструктор системы
198.      /// </summary>
199.      /// <param name="source">обозначение системы</param>
200.      /// <param name="A">первый химический элемент</param>
201.      /// <param name="B">второй химический элемент</param>
202.      /// <param name="X">общий химический элемент</param>
203.      public BinSystem(string source, Composition A, Composition B,
        Composition X)
204.      {
205.          sourceString = source;
206.          elemA = A;
207.          elemB = B;
208.          elemX = X;
209.      }
210.
211.      /// <summary>
212.      /// Позволяет задать параметры системы
213.      /// </summary>
214.      /// <param name="c">эмпирический параметр</param>
215.      /// <param name="m">число структурных единиц</param>
216.      /// <param name="n">координационное число</param>
217.      /// <param name="z">формальный заряд</param>
218.      public void SetData(double c, double m, double n, double z)
219.      {
220.          this.c = c;
221.          this.m = m;
222.          this.n = n;
223.          this.z = z;
224.      }
225.
226.      /// <summary>
227.      /// Позволяет получить параметры системы

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

228.          /// </summary>
229.          /// <returns>массив параметров</returns>
230.          public double[] GetData() { return new double[] { c, m, n, z,
numA, numB, numX }; }
231.
232.          /// <summary>
233.          /// Энтропия смешения
234.          /// </summary>
235.          public double Ssm(double x1)
236.          {
237.              double x2 = 1 - x1;
238.              double Skon = (-1) * kN * (x1 * Math.Log(x1) + x2 *
Math.Log(x2));
239.              double Skol = 2.7252 * x1 * x2 * (delR / R1) * 0.001;
240.
241.              return Skon + Skol;
242.          }
243.
244.          /// <summary>
245.          /// Среднее межатомное расстояние
246.          /// </summary>
247.          public double R(double x1)
248.          {
249.              if (R_const != -1)
250.                  return R_const;
251.
252.              if (r_2 == -1)
253.                  double.TryParse(elemB.Properties.Find(x => x.First.Second
== symbols[0]).Second[numB], out r_2);
254.              if (r_3 == -1)
255.                  double.TryParse(elemX.Properties.Find(x => x.First.Second
== symbols[0]).Second[numX], out r_3);
256.              if (r_1 == -1)
257.                  double.TryParse(elemA.Properties.Find(x => x.First.Second
== symbols[0]).Second[numA], out r_1);
258.
259.              if (r_1 != -1 && r_2 != -1 && r_3 != -1)
260.                  return x1 * r_1 + (1 - x1) * r_2 + r_3;
261.              else
262.                  return -1;
263.          }
264.

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

265.      /// <summary>
266.      /// Разность радиусов
267.      /// </summary>
268.      public double delR
269.      {
270.          get
271.          {
272.              if (r1 != -1 && r2 != -1)
273.                  return Math.Abs(r_1 - r_2);
274.              else
275.                  return -1;
276.          }
277.      }
278.
279.      /// <summary>
280.      /// Степень ионности
281.      /// </summary>
282.      /// <param name="i">Флаг: 1 - элемент А, 2 - элемент В</param>
283.      public double Eps(int i)
284.      {
285.
286.          Composition temp;
287.          int num;
288.
289.          switch (i)
290.          {
291.              case 1:
292.                  temp = elemA;
293.                  num = numA;
294.                  break;
295.              case 2:
296.                  temp = elemB;
297.                  num = numB;
298.                  break;
299.              default:
300.                  return 0;
301.          }
302.
303.          if (double.TryParse(temp.Properties.Find(x => x.First.Second
304.              == symbols[1]).Second[num], out double k)
305.              && double.TryParse(elemX.Properties.Find(x =>
306.                  x.First.Second == symbols[1]).Second[numX], out double j))

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

305.         return 1 - (z / n) * Math.Exp((k - j) * (k - j) * -0.25);
306.     else
307.         return -1;
308.     }
309.
310.     /// <summary>
311.     /// Разность степеней ионности
312.     /// </summary>
313.     public double delEps
314.     {
315.         get
316.         {
317.             if (deleps < 0)
318.                 deleps = Math.Abs(Eps(1) - Eps(2));
319.
320.             return deleps;
321.         }
322.
323.         set
324.         {
325.             deleps = value;
326.         }
327.     }
328.
329.     /// <summary>
330.     /// Теплота смешения
331.     /// </summary>
332.     public double Hsm(double x1)
333.     {
334.         if (zX == -1)
335.             double.TryParse(elemX.Properties.Find(x => x.First.Second
336.                 == symbols[2]).Second[numX], out zX);
337.
338.         return x1 * (1 - x1) * (322 * A / R(x1) * (delEps * delEps) +
339.             c * m * n * z * zX * (delR / R(x1) * delR / R(x1)));
340.     }
341.
342.     /// <summary>
343.     /// Свободная энергия Гиббса
344.     /// </summary>
345.     /// <param name="T">температура</param>
346.     public double Gsm(double x1, double T)

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата


```

345.        {
346.            return Hsm(x1) - T * Ssm(x1);
347.        }
348.
349.        /// <summary>
350.        /// Критическая температура
351.        /// </summary>
352.        public double Tmax
353.        {
354.            get
355.            {
356.                if (zX == -1)
357.                    double.TryParse(elemX.Properties.Find(x =>
358.                        x.First.Second == symbols[2]).Second[numX], out zX);
359.                return (c * m * n * z * zX * ((delR / Math.Min(R1, R2)) *
360.                    (delR / Math.Min(R1, R2)))) / (2 * kN);
361.            }
362.        }
363.        /// <summary>
364.        /// Возвращает обозначение системы
365.        /// </summary>
366.        /// <returns>обозначение системы</returns>
367.        public override string ToString()
368.        {
369.            return sourceString;
370.        }
371.
372.        /// <summary>
373.        /// Создает копию системы
374.        /// </summary>
375.        /// <returns>копия данной системы</returns>
376.        public BinSystem Clone()
377.        {
378.            BinSystem toClone = new BinSystem(sourceString, new
379.                Composition(elemA.Name, elemA.DataTable, elemA.Properties), new
380.                Composition(elemB.Name, elemB.DataTable, elemB.Properties),
381.                new Composition(elemX.Name, elemX.DataTable,

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

382.             toClone.numB = numB;
383.             toClone.numX = numX;
384.             toClone.A = A;
385.
386.             return toClone;
387.         }
388.     }
389. }
390.

```

- **Composition.cs**

```

1. using System;
2. using System.Collections.Generic;
3. using System.Data;
4. using System.Xml.Serialization;
5.
6. namespace Activision_Mendelejev_table.HelperClasses
7. {
8.     /// <summary>
9.     /// Класс, представляющий собой соединение (элемент)
10.    /// </summary>
11.    [Serializable]
12.    public class Composition
13.    {
14.        /// <summary>
15.        /// Название соединения (элемента)
16.        /// </summary>
17.        [XmlElement("Name")]
18.        public string Name { get; set; }
19.
20.        /// <summary>
21.        /// Лист свойств и соответствующих их значений
22.        /// </summary>
23.        [XmlArray("Properties"), XmlArrayItem("Property")]
24.        public List<Pair<Pair<string, string>, List<string>>> Properties { get;
25.            set; }
26.
27.        /// <summary>
28.        /// Таблица свойств элементов
29.        /// </summary>
30.        [XmlElement("Table")]

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

30.         public DataTable DataTable { get; set; }
31.
32.         /// <summary>
33.         /// Создает объект типа Composition и инициализирует лист свойств
34.         /// </summary>
35.         public Composition() { Properties = new List<Pair<Pair<string, string>,
List<string>>>(); }
36.
37.         /// <summary>
38.         /// Создает объект типа Composition и инициализирует все свойства
          класса
39.         /// </summary>
40.         /// <param name="name">название соединения(элемента)</param>
41.         /// <param name="data">таблица свойств элементов</param>
42.         /// <param name="prop">лист свойств и соответствующих им
          значений</param>
43.         public Composition(string name, DataTable data = null,
List<Pair<Pair<string, string>, List<string>>> prop = null)
44.         {
45.             Properties = prop;
46.             DataTable = data;
47.             Name = name;
48.             if (data == null)
49.                 DataTable = new DataTable() { TableName = name};
50.             if (prop == null)
51.                 Properties = new List<Pair<Pair<string, string>,
List<string>>>();
52.         }
53.     }
54. }
55.

```

- **DataGridHelper.cs**

```

1. using System;
2. using System.Data;
3. using System.IO;
4. using System.Windows;
5. using System.Windows.Controls;
6. using System.Windows.Data;
7. using System.Windows.Media;
8. using System.Xml.Serialization;

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```
9.
10. namespace Activision_Mendeleyev_table.HelperClasses
11. {
12.     /// <summary>
13.     /// Вспомогательный статический класс для работы с DataGridView
14.     /// </summary>
15.     public static class DataGridViewHelper
16.     {
17.         /// <summary>
18.         /// Вспомогательный метод для получения ячейки таблицы DataGridView
19.         /// </summary>
20.         /// <typeparam name="T">Visual type</typeparam>
21.         /// <param name="parent">предок</param>
22.         /// <returns>потомок</returns>
23.         private static T GetVisualChild<T>(Visual parent) where T : Visual
24.         {
25.             try
26.             {
27.                 T child = default(T);
28.                 int numVisuals = VisualTreeHelper.ChildrenCount(parent);
29.                 for (int i = 0; i < numVisuals; i++)
30.                 {
31.                     Visual v = (Visual)VisualTreeHelper.GetChild(parent, i);
32.                     child = v as T;
33.                     if (child == null)
34.                         child = GetVisualChild<T>(v);
35.
36.                     if (child != null)
37.                         break;
38.                 }
39.                 return child;
40.             }
41.             catch (Exception ex)
42.             {
43.                 MessageBox.Show("Ошибка получения данных из таблицы функция  
(GetVisualChild<T>)!\n\n" + ex.Message, "Ошибка", MessageBoxButton.OK,
44.                     MessageBoxImage.Error);
45.                 return null;
46.             }
47.         }
48.
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

49.      /// <summary>
50.      /// Получает строку из DataGridView
51.      /// </summary>
52.      /// <param name="index">номер строки</param>
53.      /// <param name="dg">DataGridView</param>
54.      /// <returns>Строки DataGridView</returns>
55.      private static DataGridViewRow GetRow(int index, DataGridView dg)
56.      {
57.          try
58.          {
59.              DataGridViewRow row =
60.              (DataGridViewRow)dg.ItemContainerGenerator.ContainerFromIndex(index);
61.              if (row == null)
62.              {
63.                  dg.UpdateLayout();
64.                  dg.ScrollIntoView(dg.Items[index]);
65.                  row =
66.                  (DataGridViewRow)dg.ItemContainerGenerator.ContainerFromIndex(index);
67.              }
68.              return row;
69.          }
70.          catch (Exception ex)
71.          {
72.              MessageBox.Show("Ошибка получения данных из таблицы функция
73.              (GetRow)!\n\n" + ex.Message, "Ошибка", MessageBoxButtons.OK,
74.              MessageBoxIcon.Error);
75.              return null;
76.          }
77.      }
78.      /// <summary>
79.      /// Добавляет текстовый столбец в таблицу
80.      /// </summary>
81.      /// <param name="dg">DataGridView</param>
82.      /// <param name="dat">таблица данных</param>
83.      /// <param name="f">флаг: true - соединение(элемент), false -
84.      /// система</param>
85.      /// <returns>Можно ли удалять столбцы?</returns>
86.      public static bool AddColumn(ref DataGridView dg, ref DataTable dat, bool f
87.      = true)
88.      {

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

85.         ColumnAddWindow form = new ColumnAddWindow();
86.         form.ShowDialog();
87.         try
88.         {
89.             if (form.name != "")
90.             {
91.                 if (form.symbol != "" && form.symbol != " ")
92.                     foreach (DataColumn v in dat.Columns)
93.                         if (v.Caption == form.symbol)
94.                             throw new DuplicateNameException();
95.
96.                 DataColumn col = new DataColumn(form.name) { Caption =
form.symbol };
97.                 dat.Columns.Add(col);
98.                 dg.Columns.Add(new DataGridTextColumn()
99.                 {
100.                     Header = (form.symbol == "" || form.symbol == "
") ? form.name : form.name + ", " + form.symbol,
101.                     Binding = new Binding "[" + dg.Columns.Count +
"]")
102.                 });
103.             }
104.         }
105.         catch (DuplicateNameException)
106.         {
107.             MessageBox.Show("Столбец с данным именем(обозначением)
уже принадлежит данной таблице!", "Ошибка", MessageBoxButton.OK,
MessageBoxImage.Error);
108.         }
109.
110.         if (f && dat.Columns.Count > 0 || !f && dat.Columns.Count >
1)
111.             return true;
112.         return false;
113.     }
114.
115.     /// <summary>
116.     /// Получает ячейку из DataGrid
117.     /// </summary>
118.     /// <param name="row">номер строки</param>
119.     /// <param name="column">номер столбца</param>
120.     /// <param name="dg">DataGrid</param>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```
121.          /// <returns>Ячейка DataGrid</returns>
122.      public static DataGridCell GetCell(int row, int column, DataGrid
        dg)
123.      {
124.          try
125.          {
126.              DataGridView rowContainer = GetRow(row, dg);
127.
128.              if (rowContainer != null)
129.              {
130.
131.                  System.Windows.Controls.Primitives.DataGridCellsPresenter presenter =
                  GetVisualChild<System.Windows.Controls.Primitives.DataGridCellsPresenter>(rowCo
                  ntainer);
132.
133.                  DataGridCell cell =
                  (DataGridCell)presenter.ItemContainerGenerator.ContainerFromIndex(column);
134.                  if (cell == null)
135.                  {
136.                      dg.ScrollIntoView(rowContainer,
137.                      dg.Columns[column]);
138.                      cell =
139.                      (DataGridCell)presenter.ItemContainerGenerator.ContainerFromIndex(column);
140.                  }
141.                  return cell;
142.              }
143.              return null;
144.          }
145.          catch (Exception ex)
146.          {
147.              MessageBox.Show("Ошибка получения данных из таблицы
148.              функция (GetCell)!\n\n" + ex.Message, "Ошибка", MessageBoxButton.OK,
149.              MessageBoxImage.Error);
150.              return null;
151.          }
152.      }
153.
154.      /// <summary>
155.      /// Закрашивает ячейку DataGrid
156.      /// </summary>
157.      /// <param name="row">номер строки</param>
158.      /// <param name="column">номер столбца</param>
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

154.          /// <param name="color">цвет для закрашивания</param>
155.          /// <param name="dg">DataGrid</param>
156.          public static void BrushCell(int row, int column, Brush color,
            DataGridView dg)
157.          {
158.              DataGridViewCell cell = GetCell(row, column, dg);
159.              cell.Background = color;
160.              cell.BorderBrush = color;
161.          }
162.
163.          /// <summary>
164.          /// Метод сериализации соединений/элементов/систем соединений
165.          /// </summary>
166.          /// <typeparam name="T">тип: List<Composition> или
            List<DataTable></typeparam>
167.          /// <param name="name">имя файла</param>
168.          /// <param name="data">лист соединений/элементов/систем
            соединений</param>
169.          public static void Serialize<T>(string name, ref
            System.Collections.Generic.List<T> data)
170.          {
171.              using (FileStream fs = new FileStream(name, FileMode.Create,
                FileAccess.Write))
172.              {
173.                  XmlSerializer serializer = new
                XmlSerializer(data.GetType());
174.                  serializer.Serialize(fs, data);
175.              }
176.          }
177.
178.          /// <summary>
179.          /// Метод десериализации соединений/элементов/систем соединений
180.          /// </summary>
181.          /// <typeparam name="T">тип: List<Composition> или
            List<DataTable></typeparam>
182.          /// <param name="name">имя файла</param>
183.          /// <param name="data">лист соединений/элементов/систем
            соединений</param>
184.          /// <returns>Удалось ли десериализовать?</returns>
185.          public static bool Deserialize<T>(string name, ref
            System.Collections.Generic.List<T> data)
186.          {

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата


```
187.             if (!File.Exists(name))
188.                 return false;
189.
190.             try
191.             {
192.                 using (FileStream fs = new FileStream(name,
193.                 FileMode.Open, FileAccess.Read))
194.                 {
195.                     XmlSerializer serializer = new
196.                     XmlSerializer(data.GetType());
197.                     data =
198.                     (System.Collections.Generic.List<T>)serializer.Deserialize(fs);
199.                 }
200.                 return true;
201.             }
202.             catch (FileLoadException)
203.             {
204.                 return false;
205.             }
206.
```

- **HotKey.cs**

```
1. using System;
2. using System.Collections.Generic;
3. using System.Diagnostics;
4. using System.Runtime.InteropServices;
5. using System.Windows.Input;
6. using System.Windows.Interop;
7.
8. namespace Activision_Mendelejev_table.HelperClasses
9. {
10.     /// <summary>
11.     /// Класс для привязки клавиш к действиям
12.     /// </summary>
13.     public class HotKey : IDisposable
14.     {
15.         /// <summary>
16.         /// Словарь id и горячих клавиш для обратного вызова
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```
17.      ///Слово для генерации сообщений
28.      ///Флаг для управления методами Dispose() и Dispose(bool)
33.      ///Клавиша
38.      ///Модификаторы
43.      ///Метод, который вызывается по нажатию сочетания клавиш
48.      ///Уникальный идентификатор сочетания клавиш
53.      ///
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

57.      /// Создает объект типа HotKey и регистрирует переданное сочетание
        клавиш (по умолчанию)
58.      /// </summary>
59.      /// <param name="k">клавиша</param>
60.      /// <param name="keyModifiers">модификаторы</param>
61.      /// <param name="action">метод</param>
62.      /// <param name="register">Нужно ли регистрировать?</param>
63.      public HotKey(Key k, KeyModifier keyModifiers, Action<HotKey> action,
        bool register = true)
64.      {
65.          Key = k;
66.          KeyModifiers = keyModifiers;
67.          Action = action;
68.          if (register)
69.              Register();
70.      }
71.
72.      /// <summary>
73.      /// Регистрирует сочетание клавиш
74.      /// </summary>
75.      /// <returns>Зарегистрирован или нет?</returns>
76.      public bool Register()
77.      {
78.          int virtualKeyCode = KeyInterop.VirtualKeyFromKey(Key);
79.          Id = virtualKeyCode + ((int)KeyModifiers * 0x10000);
80.          bool result = RegisterHotKey(IntPtr.Zero, Id, (uint)KeyModifiers,
        (uint)virtualKeyCode);
81.
82.          if (_dictHotKeyToCallbackProc == null)
83.          {
84.              _dictHotKeyToCallbackProc = new Dictionary<int, HotKey>();
85.              ComponentDispatcher.ThreadFilterMessage += new
        ThreadMessageEventHandler(ComponentDispatcher.ThreadFilterMessage);
86.          }
87.
88.          _dictHotKeyToCallbackProc.Add(Id, this);
89.
90.          Debug.Print(result.ToString() + ", " + Id + ", " + virtualKeyCode);
91.          return result;
92.      }
93.
94.      /// <summary>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

95.         /// Утилизирует управляемые ресурсы
96.         /// </summary>
97.         public void Unregister()
98.         {
99.             HotKey hotKey;
100.             if (_dictHotKeyToCallBackProc.TryGetValue(Id, out hotKey))
101.                 UnregisterHotKey(IntPtr.Zero, Id);
102.         }
103.
104.         private static void ComponentDispatcherThreadFilterMessage(ref
MSG msg, ref bool handled)
105.         {
106.             if (!handled && msg.message == WmHotKey)
107.             {
108.                 HotKey hotKey;
109.                 if
(_dictHotKeyToCallBackProc.TryGetValue((int)msg.wParam, out hotKey))
110.                 {
111.                     if (hotKey.Action != null)
112.                         hotKey.Action.Invoke(hotKey);
113.                     handled = true;
114.                 }
115.             }
116.         }
117.
118.         /// <summary>
119.         /// Метод IDisposable, вызывает наш virtual Dispose(bool)
120.         /// </summary>
121.         public void Dispose()
122.         {
123.             Dispose(true);
124.             GC.SuppressFinalize(this);
125.         }
126.
127.         /// <summary>
128.         /// Утилизирует ресурсы
129.         /// </summary>
130.         /// <param name="disposing">Можно ли утилизировать управляемые
ресурсы?</param>
131.         protected virtual void Dispose(bool disposing)
132.         {
133.             if (!this._disposed)

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

134.         {
135.             if (disposing)
136.                 Unregister();
137.
138.             _disposed = true;
139.         }
140.     }
141. }
142.
143.     /// <summary>
144.     /// Модификаторы
145.     /// </summary>
146.     [Flags]
147.     public enum KeyModifier
148.     {
149.         None = 0x0000,
150.         Alt = 0x0001,
151.         Ctrl = 0x0002,
152.         NoRepeat = 0x4000,
153.         Shift = 0x0004,
154.         Win = 0x0008
155.     }
156. }
157.

```

- **MathParser.cs**

```

1. using System;
2. using System.Collections.Generic;
3. using System.Data;
4. using System.Globalization;
5. using System.Text.RegularExpressions;
6.
7. namespace Activision_Mendelejev_table.HelperClasses
8. {
9.     /// <summary>
10.    /// Класс, вычисляющей значение формулы по введенной строке по определенным
    /// правилам
11.    /// </summary>
12.    public static class MathParser
13.    {
14.        /// <summary>

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

15.      /// Флаг: true - соединений, false - система
16.      /// </summary>
17.      public static bool f = true;
18.
19.      /// <summary>
20.      /// Парсинг строки для вычисления значения формулы, если данная строка
        является формулой
21.      /// </summary>
22.      /// <param name="str">содержание ячейки таблицы</param>
23.      /// <param name="dat">таблица, в которой производим вычисление</param>
24.      /// <param name="u">номер строки</param>
25.      /// <returns>результат парсинга</returns>
26.      public static double Parse(string str, ref DataTable dat, int u)
27.      {
28.          str = str.Replace(',', '.');
29.          string left = "", right = "", inner = "";
30.          int i = 0, j = 0, k = 0;
31.
32.          // Парсинг функций
33.          string[] func = { "sin", "cos", "tan", "ctan", "abs", "exp", "ln",
        "min", "max", "pow" };
34.          for (i = 0; i < func.Length; i++)
35.          {
36.              k = str.IndexOf(func[i]);
37.              if (k >= 0)
38.              {
39.                  left = str.Substring(0, k);
40.                  k += func[i].Length;
41.                  j = 0;
42.                  bool f = false;
43.                  right = "";
44.                  inner = "";
45.                  while (k < str.Length)
46.                  {
47.                      if (!f && str[k] == '(' && k < str.Length)
48.                      {
49.                          j++;
50.                          if (j == 1)
51.                          {
52.                              left += right;
53.                              right = "";
54.                              k++;

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

55.         }
56.     }
57.
58.     right += str[k];
59.
60.     if (!f && k < str.Length - 1 && str[k + 1] == ' ')
61.     {
62.         j--;
63.         if (j == 0)
64.         {
65.             inner += right;
66.             right = "";
67.             f = true;
68.             k++;
69.         }
70.     }
71.
72.     k++;
73. }
74.
75.     switch (i)
76.     {
77.         case 0:
78.             return Parse(left + Math.Sin(Parse(inner, ref dat,
79. u)) + right, ref dat, u);
80.
81.         case 1:
82.             return Parse(left + Math.Cos(Parse(inner, ref dat,
83. u)) + right, ref dat, u);
84.
85.         case 2:
86.             return Parse(left + Math.Tan(Parse(inner, ref dat,
87. u)) + right, ref dat, u);
88.
89.         case 3:
90.             return Parse(left + 1.0 / Math.Tan(Parse(inner, ref
91. dat, u)) + right, ref dat, u);
92.
93.         case 4:
94.             return Parse(left + Math.Abs(Parse(inner, ref dat,
95. u)) + right, ref dat, u);
96.     }
97. }
98.
99.
100.

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

92.             case 5:
93.                 return Parse(left + Math.Exp(Parse(inner, ref dat,
94.                     u)) + right, ref dat, u);
95.
96.             case 6:
97.                 return Parse(left + Math.Log(Parse(inner, ref dat,
98.                     u)) + right, ref dat, u);
99.
100.            case 7:
101.                int a = 0, b = 0, v = 0, g = 0;
102.                string inleft = "";
103.                while (inner[g] != ';' || a != 0 || b != 0 ||
104.                    v != 0)
105.                {
106.                    inleft += inner[g];
107.                    if (inner[g] == '(')
108.                        a++;
109.                    else if (inner[g] == ')')
110.                        a--;
111.                    else if (inner[g] == '[')
112.                        b++;
113.                    else if (inner[g] == ']')
114.                        b--;
115.                    else if (inner[g] == '{')
116.                        v++;
117.                    else if (inner[g] == '}')
118.                        v--;
119.                    g++;
120.                }
121.
122.                string inright = inner.Substring(g + 1);
123.                return Parse(left + Math.Min(Parse(inleft,
124.                    ref dat, u), Parse(inright, ref dat, u)) + right, ref dat, u);
125.
126.            case 8:
127.                a = 0; b = 0; v = 0; g = 0;
128.                inleft = "";
129.                while (inner[g] != ';' || a != 0 || b != 0 ||
130.                    v != 0)
131.                {
132.                    inleft += inner[g];
133.                    if (inner[g] == '(')

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата


```
129.             a++;
130.             else if (inner[g] == ')')
131.                 a--;
132.             else if (inner[g] == '[')
133.                 b++;
134.             else if (inner[g] == ']')
135.                 b--;
136.             else if (inner[g] == '{')
137.                 v++;
138.             else if (inner[g] == '}')
139.                 v--;
140.             g++;
141.         }
142.
143.         inright = inner.Substring(g + 1);
144.         return Parse(left + Math.Max(Parse(inleft,
145.             ref dat, u), Parse(inright, ref dat, u)) + right, ref dat, u);
146.
147.         case 9:
148.             a = 0; b = 0; v = 0; g = 0;
149.             inleft = "";
150.             while (inner[g] != ';' || a != 0 || b != 0 ||
151.                 v != 0)
152.             {
153.                 inleft += inner[g];
154.                 if (inner[g] == '(')
155.                     a++;
156.                 else if (inner[g] == ')')
157.                     a--;
158.                 else if (inner[g] == '[')
159.                     b++;
160.                 else if (inner[g] == ']')
161.                     b--;
162.                 else if (inner[g] == '{')
163.                     v++;
164.                 else if (inner[g] == '}')
165.                     v--;
166.                 g++;
167.             }
168.
169.             inright = inner.Substring(g + 1);
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

168.                                     return Parse(left + Math.Pow(Parse(inleft,
    ref dat, u), Parse(inright, ref dat, u)) + right, ref dat, u);
169.                                     }
170.                                     }
171.                                     }
172.
173.                                     //Парсинг символа x
174.                                     Match matchFuncx = Regex.Match(str, @"(x)");
175.                                     if (matchFuncx.Groups.Count > 1)
176.                                     {
177.                                         left = str.Substring(0, matchFuncx.Index);
178.                                         right = str.Substring(matchFuncx.Index +
    matchFuncx.Length);
179.                                     return Parse(left +
    double.Parse(dat.Rows[u]["X"].ToString()) + right, ref dat, u);
180.                                     }
181.
182.                                     //Парсинг бесконечности
183.                                     Match matchinf = Regex.Match(str, @"(∞)");
184.                                     if (matchinf.Groups.Count > 1)
185.                                     {
186.                                         left = str.Substring(0, matchinf.Index);
187.                                         right = str.Substring(matchinf.Index + matchinf.Length);
188.                                         return Parse(left + "10000000000000000000" + right, ref
    dat, u);
189.                                     }
190.
191.                                     string str1 = "";
192.                                     int r = -1, c = -1;
193.                                     //Парсинг конструкций для получения значений из таблиц
194.                                     try
195.                                     {
196.                                         Match matchElem = Regex.Match(str, @"([\w\[\]\.\+|-
    \*\%\/\^\_\(\)\0123456789]*){([\w\[\]\.\+|-*\%\/\^\_\(\)\0123456789 ]*){([\[\]\d\.\+|-
    \*\%\/\^\_\(\) ]*)}{([\w\[\]\.\+|-*\%\/\^\_\(\)\0123456789 ]*)}|([\w\[\]\.\+|-
    \*\%\/\^\_\(\)\0123456789]*){([\w\[\]\.\+|-*\%\/\^\_\(\)\0123456789 ]*)}");
197.                                         if (matchElem.Groups.Count > 1)
198.                                         {
199.                                             inner =
    StringHelper.DoString(str.Substring(matchElem.Index, matchElem.Length)).Trim('
    ');
200.                                             left = str.Substring(0, matchElem.Index);

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

201.                                     right = str.Substring(matchElem.Index +
    matchElem.Length);
202.                                     LinkedList<string> strs = new LinkedList<string>();
203.                                     string symbol = "";
204.                                     k = 0;
205.                                     while (k < inner.Length)
206.                                     {
207.                                         if (inner[k] == '{')
208.                                         {
209.                                             k++;
210.                                             while (k < inner.Length && inner[k] != '}' &&
    inner[k] != '{')
211.                                             {
212.                                                 str1 += inner[k];
213.                                                 k++;
214.                                             }
215.                                             strs.AddLast(str1);
216.                                             str1 = "";
217.                                         }
218.                                         else if (inner[k] == '}')
219.                                         {
220.                                             k++;
221.                                             while (k < inner.Length && inner[k] != '}' &&
    inner[k] != '{')
222.                                             {
223.                                                 str1 += inner[k];
224.                                                 k++;
225.                                             }
226.                                             strs.AddFirst(str1);
227.                                             str1 = "";
228.                                         }
229.                                         else
230.                                         {
231.                                             symbol += inner[k];
232.                                             k++;
233.                                         }
234.                                     }
235.                                     strs.AddLast(symbol);
236.                                     strs.RemoveFirst();
237.                                     return Parse(left + Find(strs, u).ToString() + right,
    ref dat, u);
238.                                     }

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

131
RU.17701729.04.15-01 12 01-1

```
239.
240.             matchElem = Regex.Match(str, @"([\w\[\\\]\.\\+\\-
    \*\\/%\\^\\_\\(\\)0123456789]*){([\d ]*;[\d ]*)}");
241.             if (matchElem.Groups.Count > 1)
242.             {
243.                 inner =
StringHelper.DoString(str.Substring(matchElem.Index, matchElem.Length)).Trim('
    ');
244.                 left = str.Substring(0, matchElem.Index);
245.                 right = str.Substring(matchElem.Index +
    matchElem.Length);
246.                 str1 = "";
247.                 k = 0;
248.                 Composition e = null;
249.                 DataTable d = null;
250.
251.                 while (k < inner.Length)
252.                 {
253.                     if (inner[k] == '{')
254.                     {
255.                         k++;
256.                         e = MendeleevTable.Elems.Find(x => x.Name ==
    str1);
257.
258.                         if (e == null)
259.                             e = MendeleevTable.Compos.Find(x =>
    x.Name == str1);
260.
261.                         if (e == null)
262.                             d = MendeleevTable.BinarySystem.Find(x =>
    x.TableName == str1);
263.
264.                         if (d == null & e == null)
265.                             throw new Exception("Отсутствует
    соединение, элемент или система" + str1 + ".", new Exception("MyException"));
266.                         str1 = "";
267.                     }
268.                     else if (inner[k] == ';')
269.                     {
270.                         k++;
271.                         int.TryParse(str1, out r);
272.                         str1 = "";
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

273.         else if (inner[k] == '{}')
274.         {
275.             k++;
276.             int.TryParse(str1, out c);
277.             str1 = "";
278.         }
279.         else
280.         {
281.             str1 += inner[k];
282.             k++;
283.         }
284.     }
285.     if (e != null)
286.         return Parse(left + e.Properties[c].Second[r] +
287.             right, ref dat, u);
288.     }
289.
290.     matchElem = Regex.Match(str, @"([\w\[\]\.\+\-\
291.     \*\%\/\^\_\(\)\0123456789]*){([\d ]*;[\w\[\]\.\+\-\*\%\/\^\_\(\)\0123456789 ]*)}");
292.     if (matchElem.Groups.Count > 1)
293.     {
294.         inner =
295.             StringHelper.DoString(str.Substring(matchElem.Index, matchElem.Length)).Trim('
296.             ');
297.         left = str.Substring(0, matchElem.Index);
298.         right = str.Substring(matchElem.Index +
299.             matchElem.Length);
300.
301.         str1 = "";
302.         k = 0;
303.         Composition e = null;
304.         DataTable d = null;
305.
306.         while (k < inner.Length)
307.         {
308.             if (inner[k] == '{}')
309.             {
310.                 k++;
311.                 e = MendeleevTable.Elems.Find(x => x.Name ==
312.                     str1);
313.                 if (e == null)

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

133
RU.17701729.04.15-01 12 01-1

```
308.                                     e = MendeleevTable.Compos.Find(x =>
    x.Name == str1);
309.
310.                                     if (e == null)
311.                                     d = MendeleevTable.BinarySystem.Find(x =>
    x.TableName == str1);
312.
313.                                     if (d == null & e == null)
314.                                     throw new Exception("Отсутствует
    соединение, элемент или система" + str1 + ".", new Exception("MyException"));
315.                                     str1 = "";
316.                                     }
317.                                     else if (inner[k] == ';')
318.                                     {
319.                                         k++;
320.                                         int.TryParse(str1, out r);
321.                                         str1 = "";
322.                                     }
323.                                     else if (inner[k] == '}')
324.                                     {
325.                                         k++;
326.                                         if (e != null)
327.                                         {
328.                                             for (c = 0; c < e.Properties.Count; c++)
329.                                                 if (e.Properties[c].First.Second ==
    str1)
330.                                                 break;
331.                                         }
332.                                     else
333.                                         for (c = 0; c < d.Columns.Count; c++)
334.                                             if (d.Columns[c].Caption == str1)
335.                                                 break;
336.                                     str1 = "";
337.                                     }
338.                                     else
339.                                     {
340.                                         str1 += inner[k];
341.                                         k++;
342.                                     }
343.                                     }
344.                                     if (e != null)
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

345.                                     return Parse(left + e.Properties[c].Second[r] +
    right, ref dat, u);
346.                                     return Parse(left + d.Rows[r][c] + right, ref dat,
    u);
347.                                     }
348.                                     }
349.                                     catch (FormatException)
350.                                     {
351.                                         throw new FormatException(string.Format("Неверная входная
строка '{0}'", str), new Exception("MyException"));
352.                                     }
353.                                     catch (ArgumentOutOfRangeException)
354.                                     {
355.                                         throw new FormatException(string.Format("Отсутствует
строка {0} или столбец {1} у данного элемента/соединения.", r, c), new
Exception("MyException"));
356.                                     }
357.                                     catch (NullReferenceException)
358.                                     {
359.                                         throw new FormatException("Отсутствует содержание ячейки,
к которой вы обратились.", new Exception("MyException"));
360.                                     }
361.
362.                                     // Парсинг скобок
363.                                     Match brackets = Regex.Match(str, @"\[([w\d\[\\\]\.\+|-
\*\/%^\^_\(\) ]*)\]");
364.                                     if (brackets.Groups.Count > 1)
365.                                     {
366.                                         i = 0;
367.                                         j = 0;
368.                                         left = "";
369.                                         right = "";
370.                                         inner = "";
371.                                         while (i < str.Length)
372.                                         {
373.                                             if (str[i] == '(' && i < str.Length)
374.                                             {
375.                                                 j++;
376.                                                 if (j == 1)
377.                                                 {
378.                                                     left += right;
379.                                                     right = "";

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

380.                i++;
381.            }
382.        }
383.
384.        if (i < str.Length && str[i] == ')')
385.        {
386.            j--;
387.            if (j == 0)
388.            {
389.                inner += right;
390.                right = str.Substring(i + 1);
391.                return Parse(left + Parse(inner, ref dat, u)
+ right, ref dat, u);
392.            }
393.        }
394.
395.        right += str[i];
396.        i++;
397.    }
398.    if (inner != "")
399.        return Parse(left + Parse(inner, ref dat, u) + right,
ref dat, u);
400.    }
401.
402.    // Парсинг действий
403.    Match matchMulOp = Regex.Match(str,
string.Format(@"({0})\s?({1})\s?({0})\s?", RegexOptions, RegexOptions));
404.    Match matchAddOp = Regex.Match(str,
string.Format(@"({0})\s?({1})\s?({0})\s?", RegexOptions, RegexOptions));
405.    var match = (matchMulOp.Groups.Count > 1) ? matchMulOp :
(matchAddOp.Groups.Count > 1) ? matchAddOp : null;
406.    if (match != null)
407.    {
408.        left = str.Substring(0, match.Index);
409.        right = str.Substring(match.Index + match.Length);
410.        return Parse(left +
ParseAct(match).ToString(CultureInfo.InvariantCulture) + right, ref dat, u);
411.    }
412.
413.    // Парсинг числа
414.    str1 = "";
415.    for (i = 0; i < str.Length; i++)

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата


```

416.         if (str[i] != ' ')
417.             if (str[i] == '.')
418.                 str1 += ',';
419.             else
420.                 str1 += str[i];
421.         if (double.TryParse(str1, out double num))
422.             return num;
423.         else
424.             throw new FormatException(string.Format("Неверная входная
строка '{0}'", str), new Exception("MyException"));
425.     }
426.
427.     /// <summary>
428.     /// Форматная строка, соответствующая числу
429.     /// </summary>
430.     private const string RegexNum = @"[-]?[d+\.\.?d*";
431.     /// <summary>
432.     /// Форматная строка, соответствующая операциям умножения и
деления
433.     /// </summary>
434.     private const string RegexMulOp = @"[\*\%]";
435.     /// <summary>
436.     /// Форматная строка, соответствующая операциям сложения и
вычитания
437.     /// </summary>
438.     private const string RegexAddOp = @"[\+\-]";
439.
440.     /// <summary>
441.     /// Выполнение математических операций
442.     /// </summary>
443.     /// <param name="match">результаты вычисления регулярного
выражения, определяющего мат. операции</param>
444.     /// <returns>результат вычисления</returns>
445.     private static double ParseAct(Match match)
446.     {
447.         double a = double.Parse(match.Groups[1].Value,
CultureInfo.InvariantCulture);
448.         double b = double.Parse(match.Groups[3].Value,
CultureInfo.InvariantCulture);
449.
450.         switch (match.Groups[2].Value)
451.         {

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

452.         case "+":
453.             return a + b;
454.
455.         case "-":
456.             return a - b;
457.
458.         case "*":
459.             return a * b;
460.
461.         case "/":
462.             return a / b;
463.
464.         case "%":
465.             return a % b;
466.
467.         default:
468.             throw new FormatException(string.Format("Неверная
                входная строка '{0}'", match.Value), new Exception("MyException"));
469.         }
470.     }
471.
472.     /// <summary>
473.     /// Поиск значения свойства в таблицах
474.     /// </summary>
475.     /// <param name="strs">лист, включающий название
        соединения(элемента), обозначение свойства и, если необходимо, доп. свойства и
        его значение</param>
476.     /// <returns>значение свойства</returns>
477.     private static double Find(LinkedList<string> strs, int u)
478.     {
479.         LinkedListNode<string> str = strs.First;
480.         DataTable d = MendeleevTable.BinarySystem.Find(x =>
            x.TableName == str.Value);
481.         Composition e = MendeleevTable.Elems.Find(x => x.Name ==
            str.Value);
482.
483.         if (e == null)
484.             e = MendeleevTable.Compos.Find(x => x.Name == str.Value);
485.
486.         if (d == null & e == null)
487.             throw new Exception("Отсутствует соединение или элемент"
                + str.Value + ".", new Exception("MyException"));

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```
488.
489.         str = str.Next;
490.         int r = -1;
491.         if (e != null)
492.         {
493.             List<Pair<Pair<string, string>, List<string>>> data =
null;
494.             data = e.Properties;
495.
496.             Pair<Pair<string, string>, List<string>> v = data.Find(x
=> x.First.Second == str.Value);
497.
498.             if (v == null)
499.                 throw new Exception("Отсутствует свойство " +
str.Value + " у данного соединения или элемента.", new
Exception("MyException"));
500.
501.             str = str.Next;
502.
503.             if (str == null)
504.                 if (f)
505.                     return double.Parse(v.Second[v.Second.Count > u ?
u : 0]);
506.                 else
507.                     return double.Parse(v.Second[0]);
508.
509.             for (int i = 0; i < v.Second.Count; i++)
510.                 if (v.Second[i] == str.Value)
511.                     r = i;
512.
513.             if (r == -1)
514.                 throw new Exception("Отсутствует значение параметра "
+ str.Previous.Value + " равного " + str.Value + " у данного соединения или
элемента.", new Exception("MyException"));
515.
516.             v = data.Find(x => x.First.Second == strs.Last.Value);
517.
518.             if (v == null)
519.                 throw new Exception("Отсутствует параметр " +
strs.Last.Value + " у данного соединения или элемента.", new
Exception("MyException"));
520.
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

521.         return double.Parse(v.Second[r]);
522.     }
523.     int c = -1;
524.
525.     for (int i = 0; i < d.Columns.Count; i++)
526.         if (d.Columns[i].Caption == str.Value)
527.             c = i;
528.
529.     if (c == -1)
530.         throw new Exception("Отсутствует свойство " + str.Value +
531. " у данной системы соединений.", new Exception("MyException"));
532.
533.     str = str.Next;
534.
535.     if (str == null)
536.         return double.Parse(d.Rows[u][c].ToString());
537.
538.     for (int i = 0; i < d.Rows.Count; i++)
539.         if (d.Rows[i][c].ToString().Replace(',', '.', ' ') ==
540. str.Value)
541.             r = i;
542.
543.     if (r == -1)
544.         throw new Exception("Отсутствует значение параметра " +
545. str.Previous.Value + " равного " + str.Value + " у данной системы соединений.",
546. new Exception("MyException"));
547.
548.     str = str.Next;
549.     c = -1;
550.     for (int i = 0; i < d.Columns.Count; i++)
551.         if (d.Columns[i].Caption == str.Value)
552.             c = i;
553.
554.     if (c == -1)
555.         throw new Exception("Отсутствует параметр " +
556. str.Last.Value + " у данной системы соединений.", new
557. Exception("MyException"));
558.
559.     return double.Parse(d.Rows[r][c].ToString());
560. }
561. }
562. }

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

557.

- **Pair.cs**

```
1. using System;
2.
3. namespace Activision_Mendelejev_table.HelperClasses
4. {
5.     /// <summary>
6.     /// Класс, который предоставляет возможность хранить два разнородных
    объекта как единое целое
7.     /// </summary>
8.     /// <typeparam name="T">тип первого аргумента</typeparam>
9.     /// <typeparam name="U">тип второго аргумента</typeparam>
10.    [Serializable]
11.    public class Pair<T, U>
12.    {
13.        /// <summary>
14.        /// Первый аргумент
15.        /// </summary>
16.        public T First { get; set; }
17.
18.        /// <summary>
19.        /// Второй аргумент
20.        /// </summary>
21.        public U Second { get; set; }
22.
23.        /// <summary>
24.        /// Создает объект типа Pair
25.        /// </summary>
26.        public Pair() { }
27.
28.        /// <summary>
29.        /// Создает объект типа Pair и инициализирует значение аргументов
30.        /// </summary>
31.        /// <param name="first">первый аргумент</param>
32.        /// <param name="second">второй аргумент</param>
33.        public Pair(T first, U second)
34.        {
35.            First = first;
36.            Second = second;
37.        }
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```
38.     };  
39. }  
40.
```

- **Point.cs**

```
1. namespace Activision_Mendeleyev_table.HelperClasses  
2. {  
3.     /// <summary>  
4.     /// Класс, который представляет собой точку с двумя численными координатами  
5.     /// </summary>  
6.     public class Point  
7.     {  
8.         /// <summary>  
9.         /// Конструктор точки  
10.        /// </summary>  
11.        /// <param name="x">координата X</param>  
12.        /// <param name="y">координата Y</param>  
13.        public Point(double x, double y)  
14.        {  
15.            this.X = x;  
16.            this.Y = y;  
17.        }  
18.  
19.        /// <summary>  
20.        /// Свойство координаты Y  
21.        /// </summary>  
22.        public double Y { get; set; }  
23.        /// <summary>  
24.        /// Свойство координаты X  
25.        /// </summary>  
26.        public double X { get; set; }  
27.    }  
28. }  
29.
```

- **RoundConverter.cs**

```
1. using System;  
2.  
3. namespace Activision_Mendeleyev_table.HelperClasses  
4. {
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

5.      /// <summary>
6.      /// Класс, позволяющий округлять значения в таблицах
7.      /// </summary>
8.      public class RoundConverter : System.Windows.Data.IValueConverter
9.      {
10.         /// <summary>
11.         /// Округляет значение до 4 знака после запятой
12.         /// </summary>
13.         /// <param name="value">значение</param>
14.         /// <returns>округленное значение</returns>
15.         public object Convert(object value, Type targetType, object parameter,
            System.Globalization.CultureInfo culture)
16.         {
17.             if (value != null)
18.             {
19.                 var val = (double)value;
20.                 return Math.Round(val,
21.                     4).ToString(System.Globalization.CultureInfo.InvariantCulture);
22.             }
23.             return null;
24.         }
25.         public object ConvertBack(object value, Type targetType, object
            parameter, System.Globalization.CultureInfo culture)
26.         {
27.             throw new NotImplementedException("", new
            Exception("MyException"));
28.         }
29.     }
30. }
31.

```

- **Strings.cs:**
 - a. **StringHelper**

```

1.  /// <summary>
2.  /// Класс для преобразования строк по определенным правилам
3.  /// </summary>
4.  public static class StringHelper
5.  {
6.      /// <summary>
7.      /// Преобразует определённый символ в соответствующий подстроочный

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

8.      /// </summary>
9.      /// <param name="c">СИМВОЛ</param>
10.     /// <returns>ПОДСТРОЧНЫЙ СИМВОЛ</returns>
11.     private static char Substring(char c)
12.     {
13.         switch (c)
14.         {
15.             case '0':
16.                 return '0';
17.             case '1':
18.                 return '1';
19.             case '2':
20.                 return '2';
21.             case '3':
22.                 return '3';
23.             case '4':
24.                 return '4';
25.             case '5':
26.                 return '5';
27.             case '6':
28.                 return '6';
29.             case '7':
30.                 return '7';
31.             case '8':
32.                 return '8';
33.             case '9':
34.                 return '9';
35.             case '+':
36.                 return '+';
37.             case '-':
38.                 return '-';
39.             case '=':
40.                 return '=';
41.             case '(':
42.                 return '(';
43.             case ')':
44.                 return ')';
45.             case '.':
46.             case ',':
47.                 return '.';
48.             default:
49.                 return c;

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата


```

50.         }
51.     }
52.
53.     /// <summary>
54.     /// Преобразует определённый символ в соответствующий надстрочный
55.     /// </summary>
56.     /// <param name="c">СИМВОЛ</param>
57.     /// <returns>надстрочный СИМВОЛ</returns>
58.     private static char Superstring(char c)
59.     {
60.         switch (c)
61.         {
62.             case '0':
63.                 return '0';
64.             case '1':
65.                 return '1';
66.             case '2':
67.                 return '2';
68.             case '3':
69.                 return '3';
70.             case '4':
71.                 return '4';
72.             case '5':
73.                 return '5';
74.             case '6':
75.                 return '6';
76.             case '7':
77.                 return '7';
78.             case '8':
79.                 return '8';
80.             case '9':
81.                 return '9';
82.             case '+':
83.                 return '+';
84.             case '-':
85.                 return '-';
86.             case '=':
87.                 return '=';
88.             case '(':
89.                 return '(';
90.             case ')':
91.                 return ')';

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

92.         case '.':
93.         case ',':
94.             return '.'; //''';
95.         default:
96.             return c;
97.     }
98. }
99.
100.         /// <summary>
101.         /// Преобразует числовые символы после буквенных и подстрочных
        СИМВОЛОВ в подстрочные
102.         /// </summary>
103.         /// <param name="str">исходная строка</param>
104.         /// <returns>преобразованная строка</returns>
105.         public static string DoString(string str)
106.         {
107.             int i = 0;
108.             string str1 = "";
109.             while (i < str.Length)
110.             {
111.                 if (i > 0 && "0123456789".Contains(str[i].ToString()) &&
                    "0123456789) QWERTYUIOPASDFGHJKLZXCVBNMqwertyuiopasdfghjklzxcvbnm".Contains(str1[i
                    - 1].ToString()))
112.                     str1 += Substring(str[i]);
113.                 else
114.                     if (i > 1 && '.' == str[i - 1] &&
                        "0123456789) QWERTYUIOPASDFGHJKLZXCVBNMqwertyuiopasdfghjklzxcvbnm".Contains(str1[i
                        - 2].ToString()))
115.                         str1 += Substring(str[i]);
116.                     else
117.                         str1 += str[i];
118.                     i++;
119.             }
120.             return str1;
121.         }
122.     }
123.

```

b. Strings

1. /// <summary>
2. /// Вспомогательный класс для заполнения Таблицы Менделеева

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

3.      /// </summary>
4.      public class Strings
5.      {
6.          /// <summary>
7.          /// Задаёт один ряд элементов по группам и подгруппам
8.          /// </summary>
9.          public Strings(string ia = null, string iia = null, string iiaa = null,
10.             string iva = null, string va = null, string via = null, string viia = null,
11.             string viiaa_1 = null, string viiaa_2 = null, string viiaa_3 =
12.             null, string ib = null, string iib = null, string iiib = null, string ivb =
13.             null,
14.             string vb = null, string vib = null, string viib = null, string
15.             viiib = null)
16.          {
17.              Ia = ia;
18.              IIa = iia;
19.              IIIa = iiaa;
20.              IVa = iva;
21.              Va = va;
22.              VIa = via;
23.              VIIa = viia;
24.              VIIa_1 = viiaa_1;
25.              VIIa_2 = viiaa_2;
26.              VIIa_3 = viiaa_3;
27.              Ib = ib;
28.              IIb = iib;
29.              IIIb = iiib;
30.              IVb = ivb;
31.              Vb = vb;
32.              VIb = vib;
33.              VIIb = viib;
34.              VIIb = viiib;
35.          }
36.
37.          public string Ia { get; set; }
38.          public string IIa { get; set; }
39.          public string IIIa { get; set; }
40.          public string IVa { get; set; }
41.          public string Va { get; set; }
42.          public string VIa { get; set; }
43.          public string VIIa { get; set; }
44.          public string VIIa_1 { get; set; }

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

41.      public string VIIIA_2 { get; set; }
42.      public string VIIIA_3 { get; set; }
43.      public string Ib { get; set; }
44.      public string IIb { get; set; }
45.      public string IIIb { get; set; }
46.      public string IVb { get; set; }
47.      public string Vb { get; set; }
48.      public string VIb { get; set; }
49.      public string VIIb { get; set; }
50.      public string VIIIb { get; set; }
51.  }
52.

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. ГОСТ 19.201-78. Техническое задание. Требование к содержанию и оформлению. // Единая система программной документации. – Москва: Издательство стандартов, 2005.
2. ГОСТ 19.404-79. Пояснительная записка. // Единая система программной документации. – Москва: Издательство стандартов, 2005.
3. ГОСТ 19.301-78. Программа и методика испытаний. // Единая система программной документации. – Москва: Издательство стандартов, 2005.
4. ГОСТ 19.505-79. Руководство оператора. // Единая система программной документации. – Москва: Издательство стандартов, 2005.
5. ГОСТ 19.401-78. Текст программы. // Единая система программной документации. – Москва: Издательство стандартов, 2005.
6. ГОСТ 19.106-78. Требования к программным документам, выполненным печатным способом. // Единая система программной документации. – Москва: Издательство стандартов, 2005.
7. ГОСТ 19.101-77. Виды программ и программных документов. // Единая система программной документации. – Москва: Издательство стандартов, 2005.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

[illegible]

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.15-01 12 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата