

```

1 using System;
2 using System.Windows.Forms;
3 using SldWorks;
4 using System.Runtime.InteropServices;
5
6
7 namespace SW_Macro_Quad
8 {
9     public partial class Form1 : Form
10    {
11        // laufende SolidWorks Objekt, das laufende Instanz darstellt, muss ↗
12        // unter bestimmten Namen aufrufbar sein
13        // Dafür Referenz für auf ein SW-Objekt gebraucht
14        SldWorks.SldWorks swApp;
15        double breite, laenge, hoehe, radius;
16
17        public Form1()
18        {
19            InitializeComponent();
20        }
21
22        private void b_about_Click(object sender, EventArgs e)
23        {
24            MessageBox.Show("Ein kleines Makro zum Erstellen von Quadern und ↗
25                Bohrungen in diesen.");
26        }
27
28        private void contact_SW_Click(object sender, EventArgs e)
29        {
30            try
31            {
32                swApp = (SldWorks.SldWorks)Marshal.GetActiveObject ↗
33                    ("SldWorks.Application");
34            }
35            catch (Exception)
36            {
37                swApp = new SldWorks.SldWorks();
38            }
39            swApp.Visible = true;
40
41            MessageBox.Show("Verbindung zu SolidWorks hergestellt.");
42        }
43
44        private void b_create_Click(object sender, EventArgs e)
45        {
46            // Abfrage der Eingabe aus allen Textboxen.
47            // TryParse probiert den String in eine double-Variable zu ↗
48            // konvertieren.
49            // Liefert true zurück, wenn String konvertiert werden konnte.
50            // Konvertierter Wert wird in Variable kopiert durch das Wort ↗
51            out.
52            if (double.TryParse(tb_breite.Text, out breite) == false ||
53                double.TryParse(tb_laenge.Text, out laenge) == false ||

```

```

49         double.TryParse(tb_hoehe.Text, out hoehe) == false) // ||
50         //double.TryParse(tb_radius.Text, out radius) == false)
51     {
52         MessageBox.Show("Fehlerhafte Eingabe!");
53         return;
54     }
55
56     // Werte müssen von mm in m konvertiert werden.
57     breite /= 1000.0;
58     laenge /= 1000.0;
59     hoehe /= 1000.0;
60     radius /= 1000.0;
61
62     // Anlegen eines neuen Bauteiles.
63     ModelDoc2 swModel = swApp.NewPart();
64
65     // Ebene oben wird selektiert und eine Skizze wird eingefügt.
66     swModel.Extension.SelectByID("Ebene oben", "PLANE", 0, 0, 0, ➤
        false, 1, null);
67     swModel.InsertSketch2(true);
68
69     // Skizze wird in Feature konvertiert, damit diese benannt ➤
        werden kann.
70     Sketch swSketch = swModel.GetActiveSketch2();
71     Feature swFeat = (Feature)swSketch;
72     swFeat.Name = "Grundflaeche";
73
74     swModel.SketchRectangle(0, 0, 0, breite, laenge, 0, true);
75
76     Feature swFeat2 = swModel.FeatureManager.FeatureExtrusion2(true, ➤
        false, false, 0, 0, hoehe, 0, false, false, false, false, 0, ➤
        0,
77     false, false, false, false, true, true, true, 0, 0, true);
78
79     swFeat2.Name = "Quader";
80
81     swModel.SaveAsSilent("C:\\temp\\Quader_L_" + tb_laenge.Text + ➤
        "_B_" + tb_breite.Text + "_H_" + tb_hoehe.Text + ".SLDPRT", ➤
        true);
82 }
83
84 private void b_cut_Click(object sender, EventArgs e)
85 {
86     if (double.TryParse(tb_radius.Text, out radius) == false)
87     {
88         MessageBox.Show("Fehlerhafte Eingabe!");
89         return;
90     }
91     ModelDoc2 swModel = swApp.ActiveDoc;
92
93     swModel.Extension.SelectByID2("", "FACE", 0.5 * breite, hoehe, ➤
        -0.5 * laenge, false, 1, null, 0);
94

```

```
95         swModel.InsertSketch2(true);
96
97         Sketch swSketch = (Sketch)swModel.GetActiveSketch2();
98         swModel.CreateCircleByRadius2(0.5 * breite, 0.5 * laenge, 0, ➤
           radius);
99         swModel.FeatureManager.FeatureCut3(true, false, false, 1, 1, ➤
           0.0, 0.0, false, false, false, false, 0, 0, false,
100           false, false, false, false, true, true, false, false, false, ➤
           0, 0.0, false);
101
102         swModel.SaveAsSilent("C:\\temp\\Quader_L_" + tb_laenge.Text + ➤
           "_B_" +
103         tb_breite.Text + "_H_" + tb_hoehe.Text + "_cut_R_" + ➤
           tb_radius.Text + ".SLDPRT", true);
104     }
105
106 }
107 }
108
```