

```
1 using System;
2 using System.Text;
3
4 /*
5  * ## Dezimal zu Binärconverter ##
6  *
7  *      Sebastian Welscher
8  *
9  *      16.04.2018
10 *
11 */
12
13 namespace BinaryConvert
14 {
15     public class Dec2Bin
16     {
17         private static int[] bin = new int[8];
18         StringBuilder sb;
19
20         public Dec2Bin()
21         {
22             sb = new StringBuilder();
23         }
24
25         public string Convert2Binary(string dec_number)
26         {
27             int calc, mod;
28             int i = 0;
29             if (dec_number.Equals(""))
30             {
31                 return "Bitte Nummer eingeben";
32             }
33             else
34             {
35                 calc = Convert.ToInt32(dec_number);
36             }
37
38             string ausgabe;
39             Array.Clear(bin, 0, bin.Length);
40             sb.Clear();
41
42             if (calc < 0 || calc > 255)
43             {
44                 return "Zahl zu groß";
45             }
46             else
47             {
48                 do
49                 {
50                     mod = calc % 2; // prüfe auf Rest
51                     calc = calc / 2; // teile durch 2
52                     bin[i] = mod; // schreibe Ergebnis ins Array
53
54                     i++; // Zähler erhöhen
55                 } while (calc != 0);
56
57                 for (int j = bin.Length - 1; j >= 0; j--)
58                 {
59                     sb.Append(Convert.ToString(bin[j])); // Stringeinzelteile
60                     an StringBuilder übergeben
```

```
61         }
62         i = 0;
63         ausgabe = sb.ToString(); // String aufbauen
64
65         return ausgabe;
66     }
67 }
68 }
69
70 /*
71 * ## Binär zu Dezimalconverter ##
72 *
73 * Sebastian Welscher
74 *
75 * 12.05.2018
76 *
77 */
78
79 public class Bin2Dec
80 {
81     private char[] chardata = new char[8];
82     private int[] intdata = new int[8];
83
84     public Bin2Dec()
85     {
86         //Konstruktor
87     }
88
89     public string Convert2Dec(string input)
90     {
91         string output;
92         ClearArray(chardata); //Arrayinhalt löschen
93         ClearArray(intdata);
94         FillCharArray(input); //CharArray füllen
95         IntoIntArray(); //CharArrayinhalte in IntArrayinhalte
96         //konvertieren
97         output = ParseArray(); //IntArrayinhalt auf 0 und 1
98         //abfragen
99         return output;
100     }
101
102     private void ClearArray(Array arr)
103     {
104         Array.Clear(arr, 0, arr.Length);
105     }
106
107     private void FillCharArray(string data)
108     {
109         chardata = data.ToCharArray(); // String in einzelne Chars
110         //aufspalten und in Array speichern
111     }
112
113     private void IntoIntArray()
114     {
115         int dat;
116         for(int i = 0; i < chardata.Length; i++)
117         {
118             dat = (int)char.GetNumericValue(chardata[i]); // Numerischen
119             //Inhalt aus Char abrufen und in int umwandeln
120             intdata[i] = dat; // Daten in
121             //IntArray speichern
122         }
123     }
124 }
```

```
117     }
118 }
119
120 private string ParseArray()
121 {
122     // To-Do: Größere Zahlen abfangen
123
124     string dat;
125     int[] convnumbers = { 128, 64, 32, 16, 8, 4, 2, 1}; //
126     // Konvertierungsnummern
127     int sum = 0, calc = 0; // Hilfsvariablen
128     for(int i = 0; i<intdata.Length;i++)
129     {
130         if (intdata[i] == 1) // wenn IntArrayeintrag == 1,
131             dann calc die jeweilige Konvertierungsnummer zuweisen
132         {
133             calc = convnumbers[i];
134         }
135         else
136         {
137             calc = 0; // ansonsten 0 setzten
138         }
139         sum = sum + calc; // Konvertierungsmummern zusammenrechnen
140     }
141     dat = sum.ToString(); // int zu string konvertieren
142     return dat;
143 }
144
145 /*
146  * ## Dezimal zu Hexadezimalconverter ##
147  *
148  * Sebastian Welscher
149  *
150  * 13.05.2018
151  *
152  */
153
154 public class Dez2Hexa
155 {
156     private int[] hexval;
157     public Dez2Hexa()
158     {
159         //Konstruktor
160         hexval = new int[2];
161     }
162
163     public string Convert2Hex(string input)
164     {
165         string output;
166
167         output = Calculate(input); // Funktion Calculate aufrufen
168
169         return output;
170     }
171
172     private void ClearArray(Array arr)
173     {
174         Array.Clear(arr, 0, arr.Length);
175     }
176 }
```

```
177     private string Calculate(string data)
178     {
179         string dat;
180         ClearArray(hexval);
181         int calc = 0, mod = 0; // Hilfsvariablen
182         if (data.Equals (""))
183         {
184             dat = "Bitte Nummer eingeben";
185         }
186         else
187         {
188             calc = Convert.ToInt32(data);
189         }
190         if(calc < 0 || calc > 255) // Testen ob Zahl kleiner 0 oder größer 255
191         {
192             dat = "Zahl zu groß";
193         }
194         else
195         {
196             mod = calc % 16; // Modulo Operation, prüfen auf Rest
197             calc = calc / 16;
198             hexval[0] = calc;
199             hexval[1] = mod;
200             dat = Convert.ToString(hexval[0] + hexval[1]);
201         }
202         return dat;
203     }
204     private string ParseNumber(int number) // Nummer überprüfen ob größer 9 und dann jeweils den passenden Buchstaben zuweisen
205     {
206         string output = "";
207         if (number < 10)
208         {
209             output = Convert.ToString(number);
210         }
211         else
212         {
213             switch (number)
214             {
215                 case 10:
216                 {
217                     output = "A";
218                 }
219                 break;
220                 case 11:
221                 {
222                     output = "B";
223                 }
224                 break;
225                 case 12:
226                 {
227                     output = "C";
228                 }
229                 break;
230                 case 13:
231                 {
232                     output = "D";
233                 }
234                 break;
```

```
235         case 14:
236             {
237                 output = "E";
238             }
239             break;
240         case 15:
241             {
242                 output = "F";
243             }
244             break;
245     }
246 }
247 return output;
248 }
249 }
250 }
251 }
252 }
```