

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace Exceptions
8 {
9     class Time
10    {
11        //DivideByZeroException
12        //IndexOutOfRangeException
13        //InvalidCastException
14
15        int hour; int min; int sec; Char ch;
16        public Time(int h, int m, int s)
17        {
18            if (h < 0 || h > 23)
19                throw new HourException("Ilegal Hour", h);
20            //throw new Exception("Ilegal Hour");
21            if (m < 0 || m > 59)
22                throw new MinException("Ilegal Minute", m);
23            if (s < 0 || s > 59)
24                throw new SecException("Ilegal Second", s);
25            this.hour = h;
26            this.min = m;
27            this.sec = s;
28            //ch = Convert.ToChar("abc");
29            int[] num = { 2, 4, 5 };
30            num[3] = 7;
31            Console.WriteLine("Conversion succeeded.");
32        }
33    }
34    class HourException : Exception
35    {
36        public int Hour;
37        public HourException(String msg, int h) : base(msg)
38        {
39            this.Hour = h;
40        }
41    }
42    class MinException : Exception
43    {
44        public int Min;
45        public MinException(String msg, int m) : base(msg)
46        {
47            this.Min = m;
48        }
49    }
50    class SecException : Exception
51    {
52        public int Sec;
```

```
53     public SecException(String msg, int s) : base(msg)
54     {
55         this.Sec = s;
56     }
57 }
58 class Program
59 {
60     static void Main(string[] args)
61     {
62         Time t;
63         try
64         {
65             try
66             {
67                 Console.WriteLine("Before");
68                 t = new Time(23, 2, 3);
69                 Console.WriteLine("After");
70             }
71             catch (HourException ex)
72             {
73                 Console.WriteLine(ex.Message);
74                 Console.WriteLine(ex.Hour);
75             }
76             catch (MinException ex)
77             {
78                 Console.WriteLine(ex.Message);
79                 Console.WriteLine(ex.Min);
80             }
81             catch (SecException ex)
82             {
83                 Console.WriteLine(ex.Message);
84                 Console.WriteLine(ex.Sec);
85             }
86             catch (DivideByZeroException e)
87             {
88                 Console.WriteLine(e.Message);
89             }
90             catch (IndexOutOfRangeException e)
91             {
92                 Console.WriteLine(e.Message);
93             }
94             catch (InvalidCastException e)
95             {
96                 Console.WriteLine(e.Message);
97             }
98             catch (Exception ex)
99             {
100                 Console.WriteLine(ex.Message);
101             }
102             finally
103             {
104                 Console.WriteLine("Always");
```

```
105     }  
106     Console.WriteLine("OK");  
107  
108     }  
109     catch (Exception ex)  
110     {  
111         Console.WriteLine(ex.Message);  
112     }  
113 }  
114 }  
115 }  
116
```