



Denne guide er oprindeligt udgivet på Eksperten.dk

## Basal type konvertering i C#

Denne artikel beskriver hvordan man konverterer mellem forskellige typer i C#.

Den forudsætter kun minimalt kendskab til C# og henvender sig primært til dem uden den store C# erfaring.

Skrevet den **05. Feb 2009** af **arne\_v** | kategorien **Programmering / C#** | ★★☆☆☆

Historie:

V1.0 - 25/04/2004 - original

V1.1 - 30/10/2004 - tilføj lidt om dansk/engelsk

### **byte/short/int/long -> string**

Brug ToString metoden eller Convert.ToString.

Eksempel:

```
int iv = 123;
string sv;
sv = iv.ToString();
eller
sv = Convert.ToString(iv);
```

### **string -> byte/short/int/long**

Brug Parse metoden eller Convert.ToXxxx.

Eksempel:

```
string sv = "123";
int iv;
iv = int.Parse(sv);
eller
iv = Convert.ToInt32(sv);
```

### **float/double -> string**

Brug ToString metoden eller Convert.ToString.

Eksempel:

```
double xv = 123.346;
string sv;
sv = xv.ToString();
eller
sv = Convert.ToString(xv);
eller
```

```
    sv = xv.ToString(new CultureInfo("en-US", false));  
eller  
    sv = xv.ToString(new CultureInfo("da-DK", false));
```

Vær opmærksom på at de to første konverteringer er locale specifik (vil give komma eller punktum afhængig af PC'ens opsætning) mens den tredje altid vil give punktum og den fjerde altid vil give komma.

### **string -> float/double**

Brug Parse metoden eller Convert.ToXxxx.

Eksempel:

```
    string sv = "123,456";  
    double xv;  
    xv = double.Parse(sv);  
eller  
    xv = Convert.ToDouble(sv);  
eller  
    xv = double.Parse(sv, new CultureInfo("en-US", false));  
eller  
    xv = double.Parse(sv, new CultureInfo("da-DK", false));
```

Vær opmærksom på at de to første konverteringen er locale specifik (og vil forvente komma eller punktum afhængig af PC'ens opsætning) mens den tredje altid vil forvente punktum og den fjerde altid vil forvente komma.

### **bool -> string**

Brug ToString metoden eller Convert.ToString.

Eksempel:

```
    bool bv = false;  
    string sv;  
    sv = bv.ToString();  
eller  
    sv = Convert.ToString(bv);
```

### **string -> bool**

Brug Parse metoden eller Convert.ToBoolean.

Eksempel:

```
    string sv = "False";  
    bool bv;  
    bv = bool.Parse(sv);  
eller  
    bv = Convert.ToBoolean(sv);
```

### **byte/short/int/long -> string (som hex)**

Brug ToString med format X eller String.Format med format X.

Eksempel:

```
int iv = 123;
string sv;
sv = iv.ToString("X");
eller
sv = String.Format("{0:X}", iv);
```

### **string (som hex) -> byte/short/int/long**

Brug Parse metoden med NumberStyles.HexNumber.

Eksempel:

```
string sv = "7B";
int iv;
iv = int.Parse(sv, NumberStyles.HexNumber);
```

### **byte/short/int/long -> float/double**

Brug simpel assignment.

Eksempel:

```
int iv = 123;
double xv;
xv = iv;
```

### **float/double -> byte/short/int/long**

Brug cast og assignment eller Convert.ToXxxx. Bemærk at der nedrundes (og decimaler tabes naturligvis).

Eksempel:

```
double xv = 123.456;
int iv;
iv = (int)xv;
eller
iv = Convert.ToInt32(xv);
```

### **byte/short/int/long -> char**

Brug cast og assignment eller Convert.ToChar.

Eksempel:

```
int iv = 65;
char cv;
cv = (char)iv;
```

```
eller
    cv = Convert.ToChar(iv);
```

### **char -> byte/short/int/long**

Brug simpel assignment.

Eksempel:

```
char cv = 'A';
int iv;
iv = cv;
```

### **byte/short/int/long -> bool**

Brug assignment af resultat af !=0 eller Convert.ToBoolean.

Eksempel:

```
int iv = 0;
bool bv;
bv = (iv != 0);
eller
bv = Convert.ToBoolean(iv);
```

### **bool -> byte/short/int/long**

Brug ?: operator eller Convert.ToXxxx.

Eksempel:

```
bool bv = false;
int iv;
iv = (bv ? 1 : 0);
eller
iv = Convert.ToInt32(bv);
```

### **DateTime -> string**

Brug ToString metoden med format.

Eksempel:

```
DateTime dt = DateTime.Now;
string sv;
sv = dt.ToString("dd-MMM-yyyy HH:mm");
eller
sv = dt.ToString("dd-MMM-yyyy HH:mm", new CultureInfo("en-US", false));
eller
sv = dt.ToString("dd-MMM-yyyy HH:mm", new CultureInfo("da-DK", false));
```

den første brugere PC'ens sprog, den anden bruger altid engelsk, den tredje bruger altid dansk.

### **string -> DateTime**

Brug Parse metoden (den gætter selv format).

Eksempel:

```
string sv = "24-Apr-2004 21:51";
DateTime dt;
dt = DateTime.Parse(sv);
eller
dt = DateTime.Parse(sv, new CultureInfo("en-US", false));
eller
dt = DateTime.Parse(sv, new CultureInfo("da-DK", false));
```

den første brugere PC'ens sprog, den anden bruger altid engelsk, den tredje bruger altid dansk.

#### **Kommentar af jesperhaun d. 11. Nov 2004 | 1**

Fin gennemgang.

#### **Kommentar af mc\_goblen d. 10. Jun 2004 | 2**

Virkelig, virkelig, virkelig lækkert artikkel når man er nybegynder inde for C#

Håber at se flere af denne slags artikler :D

#### **Kommentar af thedarkrider d. 05. Jul 2004 | 3**

Super artikel!

#### **Kommentar af mysitesolution d. 07. Feb 2006 | 4**

Fair artikel...

Måske kunne man lige pointe ud at Convert.ToX er en del langsommere end (X)blah, ikke fordi det gør det store i .Net men alligevel :)

#### **Kommentar af ranglen d. 18. Dec 2004 | 5**

ok

#### **Kommentar af visualdeveloper d. 03. Sep 2005 | 6**

nice !