

1. Casting, conversie en parsing

H4. Werken met data



Datatypes veranderen?



Datatypes veranderen

- Wat moet je doen als je 3 (integer) wil opslaan als komma-getal?
 - Eenvoudig: 3.0 (dit is **widening**)
- Wat moet je doen als je 3.643 (double) wil opslaan als geheel getal (integer)?
 - 3?
 - 4?(dit is **narrowing**)

Widening and narrowing

- **Widening:**

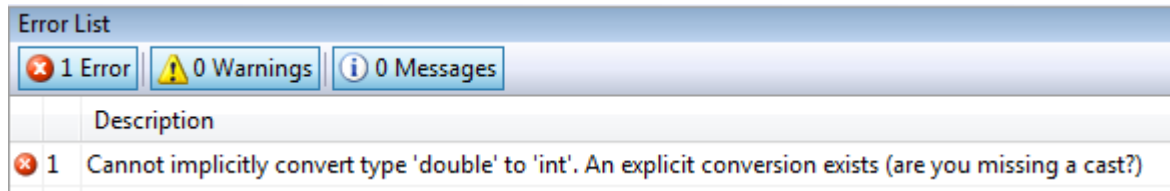
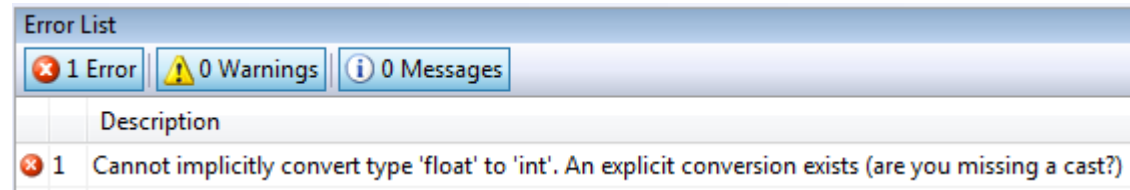
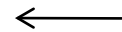
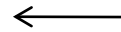
```
int i = 1;  
float x = i;
```

Werkt foutloos

- **Narrowing:**

```
float x = 1.0f;  
int i = x;
```

```
double d = 1.5;  
float i = d;
```



Oplossing: Data conversie

2,5 manieren van type conversie

Casting

- Widening en narrowing

Converteren

- Convert-bibliotheek

Parsing

- Enkel Strings omzetten (intern meestal Convert)

Casting

Manier 1

Casting (conversie)

In de informatica is een **typeconversie** (Eng.: *casting*) een operatie die een stukje data van een bepaald gegevenstype converteert naar een ander gegevenstype. (bron: wikipedia)

Enkel nodig bij NARROWING
(maar mag bij Widening)

(Casting)

Plaats het type tussen haakjes voor de variabele, literal of expressie die de compiler voor je mag omzetten

Bv

```
int getal = (int)3.7;
```


Casting in praktijk

```
double d = 1.543263463457457457457457474745356353525;  
float i = (float) d;
```



Casting: “I don’t care that this assignment could cause the loss of information. I, as the writer will take the responsibility of making sure that the program works correctly!”

Casting in praktijk II

- **Idem voor expressies** (zet hele expressie best ook tussen haakjes):
 - `int rest = (int) (3.4*2.5+4);`
- Doe je de haakjes niet rond hele expressie dan zal **enkel eerste element** gecast worden:
 - `int rest = (int) 3.4*2.5+4;`
 - Dus krijgen we error want er staat nu: `int rest= 3*2.5+4`
- **Ook** dit mag (om volgorde van casting te sturen)
 - `int rest= (int) 3.4*4+ (int) (38.66/2.2);`

Impliciet vs explicit cast



Implicit cast

Automatische widening die zonder problemen kan doorgaan

Je hoeft hier niets voor te doen (bv int to double)



(Explicit cast)

Plaats manueel het gewenste type tussen haakjes voor de expressie



Impliciete/ automatische casting

The implicit numeric conversions are:

- From sbyte to short, int, long, float, double, or decimal
- From byte to short, ushort, int, uint, long, ulong, float, double, or decimal
- From short to int, long, float, double, or decimal
- From ushort to int, uint, long, ulong, float, double, or decimal
- From int to long, float, double, or decimal
- From uint to long, ulong, float, double, or decimal
- From long to float, double, or decimal
- From ulong to float, double, or decimal
- From char to ushort, int, uint, long, ulong, float, double, or decimal
- From float to double

Casting kan mislukken

```
int i ;  
i = (int) 123456781234567890.999 ;
```

Error List

1 Error

0 Warnings

0 Messages

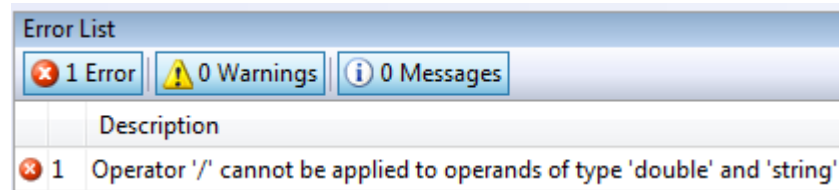
Description

1

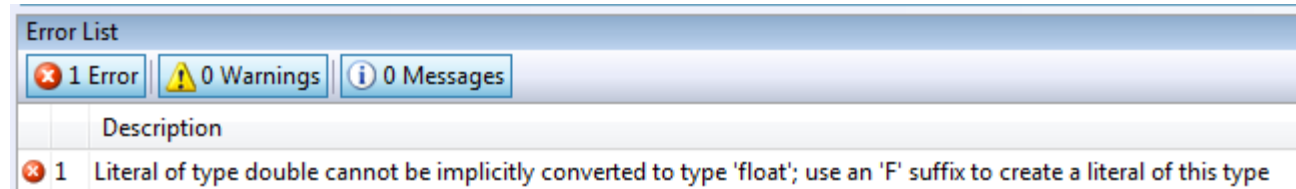
Constant value '1.23457e+017' cannot be converted to a 'int' (use 'unchecked' syntax to ov

Casting en literal waarden

```
int i ;  
i = 3.4 / "stupid" ;
```



```
float x ;  
x = 3.4 ;
```



```
float x ;  
x = 3.4f ;
```

OK

```
float x ;  
x = (float) 3.4 ;
```

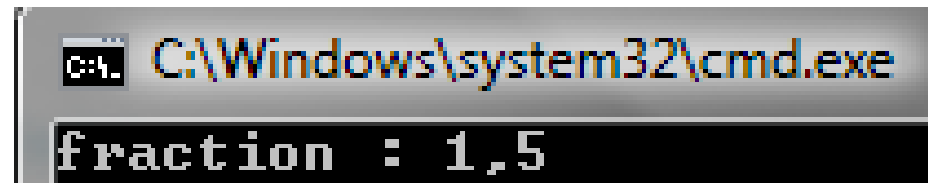
OK, gebruik makende van casting

Good practice

- Bij twijfel, beter een cast te veel doen dan een te weinig:

```
int i = 3, j = 2 ;  
float fraction ;  
fraction = (float) i / (float) j ;  
Console.WriteLine ( "fraction : " + fraction )
```

Casting naar float, garandeert float als resultaat.



C:\Windows\system32\cmd.exe
fraction : 1,5

Widening



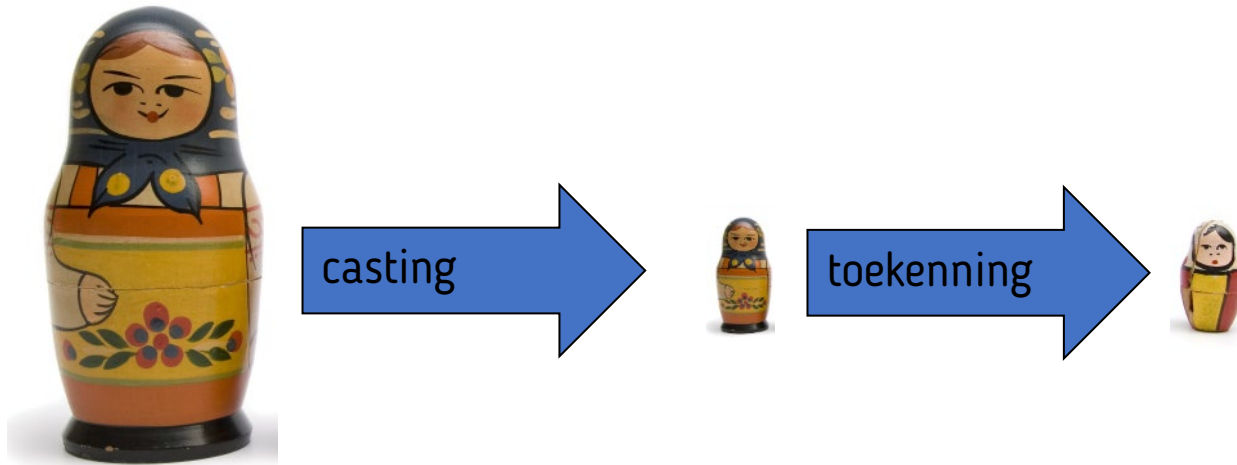
Narrowing



toekenning



Narrowing moet met casting gebeuren



Conversie

Manier 2

Convert. bibliotheek

- Ingebouwde bibliotheek met tal van omzettings-methoden.

Method	Description
<code>ToBoolean()</code>	Converts a specified value to an equivalent Boolean value
<code>ToByte()</code>	Converts a specified value to an 8-bit unsigned integer
<code>ToChar()</code>	Converts a specified value to a Unicode character
<code>ToDecimal()</code>	Converts a specified value to a decimal number
<code>ToDouble()</code>	Converts a specified value to a double-precision floating-point number
<code>ToInt16()</code>	Converts a specified value to a 16-bit signed integer
<code>ToInt32()</code>	Converts a specified value to a 32-bit signed integer
<code>ToInt64()</code>	Converts a specified value to a 64-bit signed integer
<code>ToSByte()</code>	Converts a specified value to an 8-bit signed integer
<code>ToSingle()</code>	Converts a specified value to a single-precision floating-point number
<code>ToString()</code>	Converts the specified value to its equivalent String representation
<code>ToUInt16()</code>	Converts a specified value to a 16-bit unsigned integer
<code>ToUInt32()</code>	Converts a specified value to a 32-bit unsigned integer
<code>ToUInt64()</code>	Converts a specified value to a 64-bit unsigned integer

Gebruik

Plaats tussen de ronde haken de variabele, literal of expressie die moet omgezet worden:

Voorbeeld:

- `int result= Convert.ToInt32(45.5);`
- `short otherresult= Convert.ToInt16(result);`

Parsing

Manier 3

Parsing

- Definitie:
 - “Een **parser** (van het Engelse to **parse**, ontleden, en het Latijnse pars, deel) is een computerprogramma, of component van een programma, dat de grammaticale structuur van een invoer volgens een vastgelegde grammatica ontleeft (parset). Een parser zet ingevoerde tekst om in een datastructuur.” (bron Enyclo.nl)
- Gebruikt om strings naar een specifiek data type om te zetten

datatype.Parse()

- Ieder bestaand datatype heeft ingebouwde .Parse() methode
- Methode zal proberen zo goed mogelijk de string naar het gewenste datatype om te zetten
- Bijvoorbeeld:

```
int numVal = Int32.Parse("-105");  
Console.WriteLine(numVal);
```


Demo time

- Casting
- Conversie
- Parsing

