

ConvertToLongExpression

Tue, Jan 14, 2020 [Linq](#)

ConvertToLongExpression

Table of Contents

- [ConvertToLongExpression](#)
 - [What it is](#)
 - [Code example](#)
 - [Explanation of the example](#)

What it is

ConvertToLongExpression returns an Long, as a result of a conversion from another type, for example from a string to an Long.

Before converting properties from one type to another, it may be necessary to check if the conversion is possible, in order to prevent runtime errors. When there is no doubt, the conversion can be done without checking.

- See [LINQ](#) for details about LINQ and the AI-Framework.
- Go to the category for [all LINQ expressions](#).

Compare with [CanConvertToLongExpression](#)

Code example

```
public StringProperty ExampleString => StringProperty(description: "string example");public LongProperty ExampleLong => LongProperty(description: "long example");public IMethodCall AssignLong()  
{ return Method(() => new Body { new If(ExampleString.CanConvertToLong()) { ExampleLong.Assign(ExampleString.ConvertToLong()) } });}
```

Explanation of the example

A StringProperty that contains the value "A", can **not** be converted to a Long.

A StringProperty that contains the value "123.45", can be converted to a Long. It results in the Long 123.45.

In this example, the LINQ expression in line 8 CanConvertToLong checks if the ExampleString can be converted to the type Long.

The actual conversion (ConvertToLong) is done in line 10.

Online URL:

<https://wiki-ai-framework.abstract-it.nl/article/converttolongexpression-241.html>

```
SyntaxHighlighter.config.stripBrs=true; SyntaxHighlighter.all();
```