Wt - Feature #2794

[PROJECT] Create Guidelines Subjecting Namespaces

03/11/2014 12:57 AM - I. Lazaridis

Status:	New	**Start date:**	03/10/2014
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			

Description

Code which contains the namespaces, often very nested, looks like this:

```
my::library::List<your::library::Class> 
```

This results in reduced readability.

There are two main solutions to overcome this, the "using directive" and the "using declaration"

The "using" Directive

A general **using directive** can simplify the code:

```
using namespace my::library;
using namespace your::library;
```

```
List<Class>;
```

This has 2 problems:

1. it can lead to name collisions (e.g. when two libraries have the same class names).

   1. it is still not clear where the names (functions, types, classes etc.) belong too.

The "using" Declaration

A more specific **using declaration** can solve this issue:

```
using my::library::List;
using your::library::Class;
```

```
List<Class>;
```

- The code is compact, readable and clean.
- The information for the location of "List" and "Class" is still available

And additional benefit is, that code libraries can be changed (e.g. for testing) like this:

```
using my::library::List;
//using your::library::Class;
using my::library::Class;
```

```
List<Class>;
```

Suggestion

The suggestion is to have this code guideline (especially for examples):

- use "using declarations" (
  using library::Class) instead of writing library::Class.

History

#1 - 03/11/2014 02:49 AM - I. Lazaridis

from RE: Blog Examples and Wt Best Practices

Koen Deforche wrote:

> I. Lazaridis wrote:
>
> > > We've actually moved away from importing the Wt namespace in examples to make them easier to understand (even if they are then a bit harder to read).
>
> >
>
> > I understand your point of view.
>
> >
>
> > Readability should be compared to other (non C) web frameworks, too, and "with the eyes" of a newcomer (not long-year experts, like you). Someone who is e.g. used to ruby code will get simply a shock looking at the code, backing immediately off.
>
> I understand that. But we should not only make it look easy, it should also be actually easy, and stripping of namespaces from examples turns out to make actual 'copy/pasting' harder and slows down comprehension.

"simple copy/paste" is not a priority requirement (at least that's not the reason I use C). And what slows my comprehension down, is code full of redundancy and repetition.

As a solution: you don't have to strip namespaces away, just leave them in the head section of the file (using declarations).