

Re: classes, strings, learning in VS.NET

Source: http://coding.derkeiler.com/Archive/C_CPP/comp.lang.cpp/2003-12/0261.html

From: Peter van Merkerk (*merkerk_at_deadspam.com*)

Date: 12/02/03

Date: Tue, 2 Dec 2003 16:24:13 +0100

```
> > ----- X.h -----
> > #ifndef X_H
> > #define X_H
> >
> > class X
> > {
> > public:
> > virtual void foo();
> > };
> >
> > #endif
> >
> > When the X.h file is seen once inside a translation unit the next
time
> > it gets included it will skip the definitions and declarations
inside
> > the X.h file. Most experienced C++ programmers will always put
include
> > guards in header files, even when leaving those out would not cause
> > redefinitions errors at that point in time.
>
> Since the OP is using VS, he could merely have
> #pragma once
> at the top of the headers, thus not needing to check #ifdef etc.
```

Since we are discussing standard C++ here, I prefer to stay as close to standard C++ as possible. The C++ standard does not specify what a compiler should do when it encounters `#pragma once`, other than ignoring it in case the compiler does not recognize it. Many compilers do not support this `#pragma`, and consequently putting `#pragma once` on top of the header will not result in the desired behaviour with those compilers. The solution I proposed should work on any reasonably conforming compiler.

--

Peter van Merkerk
peter.van.merkerk(at)dse.nl