Subject: c++ standard read-only property flag would be handy
Posted by luoganda on Tue, 12 Dec 2017 10:59:34 GMT
View Forum Message <> Reply to Message

for c/c++ language usually set/get are needed for properties,
using public_read section or flag would be useful for
properties, so there are not unneded code, eg:

```
class Class{
  public:
  //accessed and modifiable by any
  int someProp;

  //this would be non-modifiable in outer usage
  public_read:
  int somePropX;

  //or
  int public_read somePropX;
  //or
  public_read int somePropX;
  //or something...
};
```

for compilers that would not implement this,
just "unsafe" public: would be used instead.

Subject: Re: c++ standard read-only property flag would be handy
Posted by mirek on Wed, 13 Dec 2017 08:21:39 GMT
View Forum Message <> Reply to Message

luoganda wrote on Tue, 12 December 2017 11:59for c/c++ language usually set/get are needed for properties,
using public_read section or flag would be useful for
properties, so there are not unneded code, eg:

```
class Class{
  public:
  //accessed and modifiable by any
  int someProp;

  //this would be non-modifiable in outer usage
  public_read:
  int somePropX;

  //or
```

```
  int public_read somePropX;
  //or
  public_read int somePropX;
  //or something...
};
```

for compilers that would not implement this,
just "unsafe" public: would be used instead.

IDK, people are having these various ideas about properties, but I do not understand why. Is it really so hard to write () ?

Also, with (writable) properties, one can do things like

```
a.ink = White;
a.font = Arial;
```

which IMO is really verbose and inconvenient compared to "U++ chained modifiers" style

```
a.Ink(White()).SetFont(Arial());
```

Subject: Re: c++ standard read-only property flag would be handy
Posted by luoganda on Wed, 13 Dec 2017 11:44:27 GMT
View Forum Message <> Reply to Message

Bar::Item(orIt'sDerivates) is a good example why.
ToolButton is one such derivate.

There is no way to read properties, only Set... are there like Text(), but no Get(),
unless someone 1)subclasses it or 2)duplicate data in advance <=both is unneded

If this c++ flags were available, there would be no hokus pokus to
get those values now(noNeedToWrite Get...) and worying if both set/get is implemented,
because they would already be available by design.

+ you get direct access to it, no need to worry if
setter/getter is written as by val/ref ,like ...& Get(...) and Set(...&) to access it byRef

---

Subject: Re: c++ standard read-only property flag would be handy
Posted by mirek on Wed, 13 Dec 2017 12:54:06 GMT

View Forum Message <> Reply to Message

Bar::Item is not really a good example: there are no member variables in it...

Subject: Re: c++ standard read-only property flag would be handy
Posted by luoganda on Wed, 13 Dec 2017 17:57:09 GMT
View Forum Message <> Reply to Message

yeah, i know,
post was re-edited a few moments later...

Subject: Re: c++ standard read-only property flag would be handy
Posted by mirek on Wed, 13 Dec 2017 18:06:57 GMT
View Forum Message <> Reply to Message

Well, but we are not in control of C++ standard anyway...

Subject: Re: c++ standard read-only property flag would be handy
Posted by luoganda on Fri, 15 Dec 2017 14:15:33 GMT
View Forum Message <> Reply to Message

yeah, i know that too...