Subject: Why struct instead of class?
Posted by jeremy_c on Sat, 06 Jun 2009 04:47:28 GMT
View Forum Message <> Reply to Message

Why do the example use struct instead of class? i.e.

```
struct MyAppWindow : TopWindow {
    virtual void Paint(Draw& w) {
        w.DrawRect(GetSize(), SWhite);
        w.DrawText(20, 20, "Hello world!", Arial(30), Magenta);
    }

    MyAppWindow() {
        Title("My application").Zoomable().Sizeable();
    }
};
```

Jeremy

Subject: Re: Why struct instead of class?
Posted by cbpporter on Sat, 06 Jun 2009 15:52:43 GMT
View Forum Message <> Reply to Message

Well in modern C++ the difference between struct and class is pretty much trivial and related to default access rights: members in structs are public by default and members in classes are private by default.

C++ has a lot of duplicate features, so I tend to allays use only class to avoid it. But it makes no difference, use the one you like better, just don't mix and match in inheritance trees .

Subject: Re: Why struct instead of class?
Posted by mirek on Sun, 07 Jun 2009 19:46:33 GMT
View Forum Message <> Reply to Message

jeremy_c wrote on Sat, 06 June 2009 00:47Why do the example use struct instead of class? i.e.

```
struct MyAppWindow : TopWindow {
    virtual void Paint(Draw& w) {
        w.DrawRect(GetSize(), SWhite);
        w.DrawText(20, 20, "Hello world!", Arial(30), Magenta);
    }

    MyAppWindow() {
        Title("My application").Zoomable().Sizeable();
    }
};
```

Jeremy

Because I am lazy to write "public" for "final use" classes:)

I mean, access control is important for reusable classes. But if I am writing concrete app, I usually do not bother using access control as chances that class is going to be reused by other code is nil.

Mirek

---