brokndodge wrote on Thu, 01 April 2010 00:57
I think that is where my confusion lies. I don't understand the difference or which one I should be doing. I don't fully understand most of the Perl code I write either, but it works and I know that in this instance to get this result I can do this. Seems that with c++ I do need to fully understand it. Class member or subclass seems like splitting hairs to me. So maybe that is where the problem lies.

Hello, Brokn.

To understand new things which you didn't understand before you need to construct new meaning.
First of you can use your previous experience to do analogues. For example, word of "class" is a programming thing that have concrete meaning, but this word is new for you. I suggest to use analogue word, such as "box". In the one big box you can place some small boxes. Next step is to expand box meaning from cube form to use different forms, e.g. flexible form.
The compiler can understand inner (private), outer (public) and protected inner (protected) places of box (class). The struct by default is public for compiler, but class is private - this is a difference between class and struct in C++.
Now, when you associate your previous meaning with current you can imagine how some box (class) can be used to construct another boxes (classes/structures). For example, you have following class declarations (just templates/forms from which compiler can construct new objects):

class Shape { };
class Square : Shape { };
class Circle : Shape { };
class CircleWithSquare : Circle, Square { };

In your imagination you can "draw" following images (or hologram):

Where ordered arrows forms dependence graph (note: some associative structure in another meaning) between classes. Now, you can understand paths to access dependence classes from one to another, e.g. CircleWithSquare object (instantiated class) can access Square and Circle class structures from which it constructed, as well as Shape.

Next step is improving of your thinking, his abilities to understand (new and/or previous) things. Programming as a whole is just a part of it, but not only.

Hope, I explained to you some meaning of your "how" and "why" questions.

File Attachments
1) Shapes.png, downloaded 947 times