

Re: General Opinion on a how to?

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2007-01/msg04466.html>

- *From:* "Jamey McElveen" <jamey.mcelveen@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Tue, 30 Jan 2007 10:08:27 -0500
-

You could store the name of the algorithm in the database along with some params for it. I have done this two ways in the past successfully. First, you could store the params in a TEXT field as xml. Second you can create another detail table with key value pairs. Either way you would not have to change the database as you add new algorithm to the system. We used a system similar to this for calculating holidays for a calendar. Almost all holidays on the calendar could be calculated with a few basic algorithms.

```
<holiday name="CHRISTmas" algorithm="date" day="25" month="12"/>  
<holiday name="July 4th" algorithm="date" day="4" month="7"/>  
<holiday name="Thanksgiving" algorithm="weekday" month="11" week="4"  
weekday="5"/>
```

however easter was different so we had create a special algorithm for it.

```
<holiday name="Easter" algorithm="easter"/>  
<holiday name="Good Friday" algorithm="relative" relativeHoliday="Easter"  
metric="day" distance="-2"/>
```

All in all we have arround 5 algorithms and almost 100 holidays. This came in very usefull this year because with daylight savings time changing we had to only change the xml file.

Hope this helps

Jamey

"Daniel" <nospam@xxxxxxxxxxxxxxxx> wrote in message
[news:%23O\\$hy%237QHHA.1200@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:%23O$hy%237QHHA.1200@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx)

That part is already done and complete. The issue is more the back end storage of the prize pool break down to hard code in code as your method suggests or not?

If i have a database table that reads the variables out then that to be is the only way forward.

As Jamey said, a interface and a factory sounds a good plan and i can instatiate each from the database in advamce choosing the one needed at run time.

Re: General Opinion on a how to?

The way i see my calculation working with the aid of Jameys tips is this:

- load a game
- Retrieve Tournament placed winning positions from database as a %, top 30%, top 1% (winner takes all), etc
- Load into a prize pool object instance for that game

My biggest issue is the distribution of that prize pool. Rather than hard code, "if 3 prize positions then break down as follows", "if 4 etc".

I would rather some form of algorithm for or setting in my database to say that for this game use this ditribution, for this one use another one, and eaxh distribution be stored in a back end db table.

But how do you store a 'distribution'? I guess its an algorithm, if so i could store in the db which algorithm to use etc? this is where i am stuck....

Any ideas?

"Peter Bradley" <pbradley@xxxxxxxxxxx> wrote in message
news:%23%23qm4i7QHHA.1860@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

Seems to me you need to do some analysis.

Some obvious objects are:

Game
Player
Prize
PrizePool

Now, can you say what state variables and methods these objects should have? Does thinking about that suggest to you that you may want an inheritance hierarchy, or an interface for some of these objects? E.g.

Could there be different types of Games (Pool, Poker, Darts, Domines, Cribbage)? Would they have extra/different state and behaviour?

What about prizes? Would an interface work best here? Then a PrizePool could hold objects that implement the interface?

Will a Game hold a collection of players? Presumably a PrizePool will hold a collection of Prizes. Since the Prizes may be different, will they be different in an inheritance sense, or in an interface sense? Will a PrizePool need a reference to a Game (i.e. need to know which Game it is the PrizePool for)? Will a Prize need a reference to a Game, too? Or could the same Prize be used for many Games.

Re: General Opinion on a how to?

There's no substitute for good analysis.

HTH, IMHO, YMMV, Just my 2c etc etc

Peter

"Daniel" <nospam@xxxxxxxxxxxxxxxx> wrote in message
news:O\$8rOP6QHHA.4916@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

Hi guys

I have a requirement to calculate a prize pool for a competition similar to a poker tournament. But i am stuck on best ways or even if i have the right ideas, so:

Players pay to play, total paid is prize pool and is distributed to top 3 or so winners.

Variables are number of players who will win, if a lot enter i increase number of winners. i think a standard top 30% rounded off would work for calculating paid positions? So 10 players. 30% would be 3 of them (rounded)

Now to calculate the actual amount of prize each gets i'd like to be able to vary it. So i could offer or be versatile enough to offer different distributions. Such as, a winner takes all, a top 2 get the most and then 3rd gets a fair bit less, or a 4 out of 10 get prizes.....etc

Also i'd like to be able to offer prizes where say first place can get cash AND a car for example. But other 2 places just cash etc etc

This is where i am stuck, do i go for a hard coded table in database and read off it when my prog loads into a struct and just reference that when doing calculations? One way forward, but then if a

Re: General Opinion on a how to?

number of
players enters that i havent accounted for, then what?

Alternatively as i said above i calculate winning positions
based on a %
but then how do i decide which of those positions wins what?

Very stuck and confused, i keep thinking this must be
simple. any ideas
or pointers welcome?

Thanks