

# Re: java class hierarchy

---

*Source:* <http://coding.derkeiler.com/Archive/Java/comp.lang.java.programmer/2007-07/msg02219.html>

---

- *From:* "Oliver Wong" <[owong@xxxxxxxxxxxxxxxx](mailto:owong@xxxxxxxxxxxxxxxx)>
  - *Date:* Tue, 24 Jul 2007 14:07:56 -0400
- 

"Thomas" <[arabel9@xxxxx](mailto:arabel9@xxxxx)> wrote in message  
[news:f7td3a\\$lcs\\$2@xxxxxxxxxxxxxxxx](mailto:news:f7td3a$lcs$2@xxxxxxxxxxxxxxxx)

Hello I 'm writing a simple ONP expression evaluator with functions in java.

The constrains about the class hierarchy are like that :

abstract Symbol implements Evaluate (which contains method evaluate)

/\

/\

/\

abstract Function abstract Operand

/|\|/|

\

Plus Minus Ln Literal Const Variable

The evaluation process should be : after parsing the input we put all those object into the queue. By reading the queue and using the stack to keep the temporary values (both of them I implement on my own) i have to do like that

:

1) if it is an Operand put in on stack

2) if it is an function and

2)a we have enough arguments on stack we valuate the function

2)b we throw exeption

(i can distinguish beetwen this two types and here i don't need help, because of the diffrent number of arguments they take and exceptions)

In the main loop I have to keep the refrence to the object taken from FIFO,

and here is the problem : what type of refrence it should be ? Not a symbol

because it is abstract, but it implements the Evaluate interface. I confused. Any suggestions, please ?

## Re: java class hierarchy

There's no hard and fast rule, but as a rule of thumb, in this situation you would use the least general type which is a supertype of all the possible types you might get from your FIFO queue.

The issue is that you will probably want to do something with each element you take out of the queue. Usually, to "do something" involves invoking a method. Which means the method you want to invoke must be part of the type of the reference (or you could use casting, but that gets messy).

So say if you wanted to call the `.doSomething()` method on each element, the type of the reference would need to be such that the `.doSomething()` method is defined on that type. If the `.doSomething()` method is defined in `Symbol`, then use `Symbol` as the type of the reference. If the method is defined in `Evaluate`, then use `Evaluate`. If it's defined in both, then it doesn't matter too much which one you use.

– Oliver

.