

First shot at lisp mini-app – How to make it lispy and not lisp-c

Source: <http://coding.derkeiler.com/Archive/Lisp/comp.lang.lisp/2005-04/msg00802.html>

- *From:* "jonathon" <j_mckitrick@xxxxxxxxxxx>
 - *Date:* 13 Apr 2005 12:59:48 -0700
-

The original app is in Python with GTK as the interface. It maintains a list of accounts and their starting balances. Transactions are saved in a list in an object called the TransactionManager (TM).

At reconciliation time, each account queries the TM for transactions referring to said account. The amount is extracted from the transaction, and added to (or subtracted from) the balance. Simple.

First, in Python, when each account object is created, it receives a reference to the TransactionManager. In Lisp, should this just be a global variable that each account object calls when needed, or should it be stored in a slot within the account object? What precedent would there be for this kind of decision?

Second, what can I look for that would be an indication of too much like Python or C, where I should try to be more Lisp-ish, such as using macros, closures, etc?

.

- *Follow-Ups:*
 - ◆ ***Re: First shot at lisp mini-app – How to make it lispy and not lisp-c***
 - ◇ *From:* Kenny Tilton
 - ◆ ***Re: First shot at lisp mini-app – How to make it lispy and not lisp-c***
 - ◇ *From:* Pascal Bourguignon
- Prev by Date: ***Re: change-class on generic functions***
- Next by Date: ***Re: Is there an efficient way to get file length w/o reading file?***
- Previous by thread: ***sbcl threads semaphore?***
- Next by thread: ***Re: First shot at lisp mini-app – How to make it lispy and not lisp-c***
- Index(es):
 - ◆ ***Date***
 - ◆ ***Thread***