Homework 3

CS 482

February 17, 2005

The following is one of the simplest possible definitions.

```
class Person {
   (extent PersonExt
    kesy (SSN)
   )
   attribute Integer SSN;
   attribute String  Name;
}

class Account {
   (extent AccountExt
    keys (ActId)
   )
   attribute Integer ActId;
   relationship Set<Person> Owner;
   relationship Set<TransactionActivity> Transactions
         inverse TransactionActivity :: ActivityAccount;
}

class TransactionActivity{
   (extent TransactionActExt)
   attribute enum TransType {deposit, withdraw} Type;
   attribute float Amount;
   attribute Date  TransactionDate;
   relationship Account ActivityAccount
         inverse Account :: Transactions;
}
```

An example of an Account object:

```
(#1, [123,
      {([#p1, ''John'']), ([#p2, ''Mary''])},
      {([#t1, withdraw, 10.5, 2005-2-10, #1]),
       ([#t2, deposit, 100.0, 2005-2-11, #1])}
     ])
```

Note: The book does not clearly state how the value should be specified for a relationship. It could have been specified by the following set of objects:

```
(#1, [123, {#p1, #p2}, {#t1, #t2}])
```

```
([#p1, ''John''])
([#p2, ''Mary''])

([#t1, withdraw, 10.5, 2005-2-10, #1])
([#t2, deposit, 100.0, 2005-2-11, #1])


SELECT A.AcctId
FROM   AccountExt A, TransactionActExt T
WHERE T.Type = withdraw AND T.Amount > 10000
```