## Identify the properties that a binary relation has

A binary relation is a set where each member is a pair of values.

A ternary relation is a set where each member has three values.

A quaternary relation is a set where each member has four values.

Binary relations are very common.

Remember the ordering module? Pass it a set and it creates a bunch of operations that you can then use: first, last, next, prev, etc.

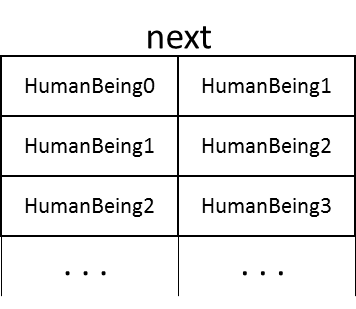
next and prev are binary relations. For example, suppose you create this set:

**sig** HumanBeing {}

and then call the ordering module with it:

**open** util/ordering[HumanBeing]

next is this binary relation:



first.next returns HumanBeing1

The ordering module creates a binary relation from the atoms in the HumanBeing set. *The* ‘*next*’ *binary relation has ordered the atoms in the set*.

Let’s create our own binary relation from the atoms in the HumanBeing set.

Let’s create a binary relation for ‘*has the same birthday*’. Roger Costello has the same birthday as Stan Efferding, so (Roger Costello, Stan Efferding) is in the binary relation. Also, Roger Costello has the same birthday as Roger Costello, so (Roger Costello, Roger Costello) is in the binary relation. Plus, (Stan Efferding, Stan Efferding) is in the binary relation. So, if HumanBeingi is in the HumanBeing set, then (HumanBeingi, HumanBeingi) must be in the binary relation.

Binary relations that contain (x, x) for every atom x in set A are so frequently encountered that such relations are said to have the *reflexive* property. Here’s the official definition:

Given a set A and a relation R in A, R is *reflexive* if and only if all the ordered pairs of the form (x, x) are in R for every x in A.

For this lab you are to create a set A, create a binary relation R on A, and then constrain R to be reflexive.

Create a **pred** and name it reflexive. Call this pred with two arguments: (1) a set, and (2) a relation. Hint: use **univ**.