

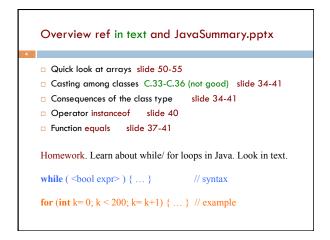
Assignment A3: Doubly linked Lists

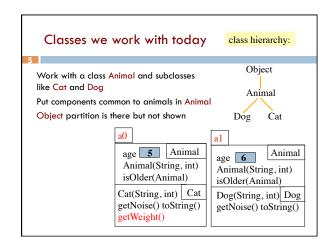
Idea: maintain a list (2, 5, 7) like this:

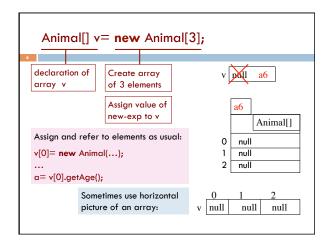
halvalue a6 v 5 v 7 succ a8 v 7
succ a6 v succ a8 v 7
succ null

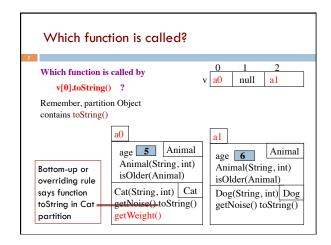
Easy to insert a node in the beginning!

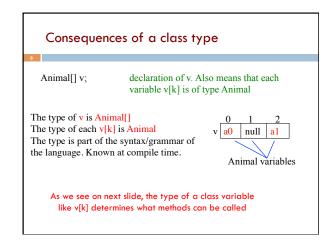
Also, if we have a variable that contains a pointer to a node, it's easy to remove that node or insert another value before or after that node.

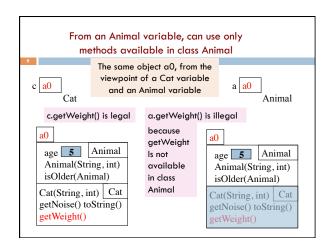


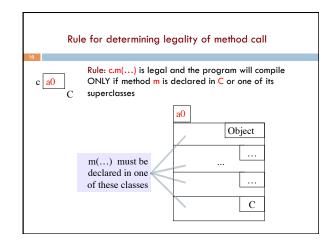


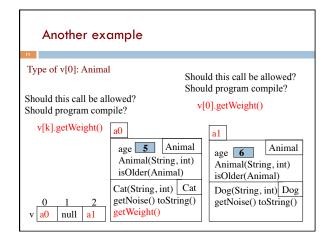


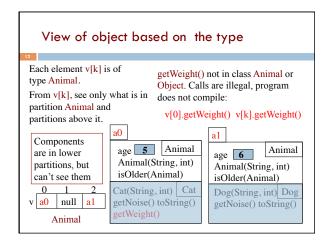


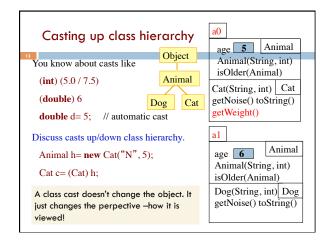


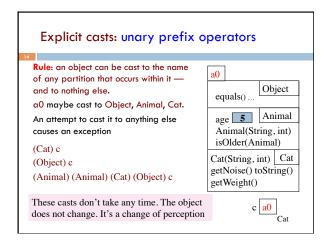


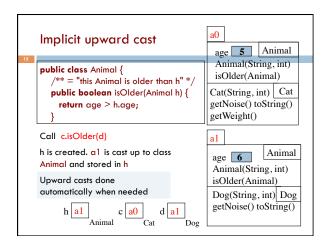


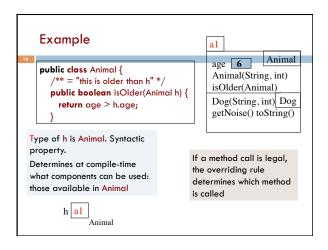


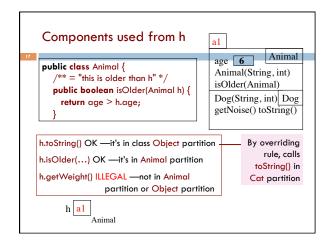


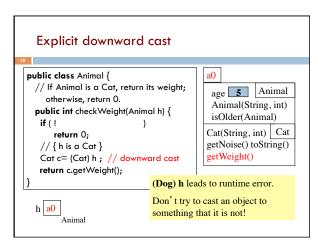


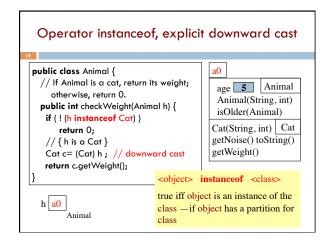


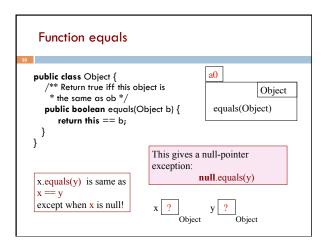












Overriding function equals Override function equals in a class to give meaning to: "these two (possibly different) objects of the class have the same values in some of their fields" For those who are mathematically inclined, like any equality function, equals should be reflexive, symmetric, and transitive. Reflexive: b.equals(b) Symmetric: b.equals(c) = c.equals(b) Transitive: if b.equals(c) and c.equals(d), then b.equals(d)

