20/04/2015 utility module.pl



utility_module.pl -- Utility Predicates 3 1



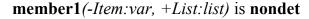




- This module has some useful list based predicates
- @author Sam Neaves

member1(+Item:atom, +List:list) is semidet

1: This is true when *Item* is in *List*.



2 : Item is the first item in the List or fails for an empty List

member1(+Item:atom, -List:list) is det

3 : *Item* is the head of a *List*, then a var.

drop(+N:int, +L:list, -NewL:list) is **det**

Means list *NewL* is *L* with its first *N* elements removed.

index of(+List:list, +Element:atom, -Index:int) is nondet

Finds the (Oth) *Index* for *Element* in *List*, will find the next *Index* on back tracking.

count of(+List:list, +Element:atom, -Count:int) is **det**

1: Count of Element in List.

count of(+List:list, +Element:atom, +Count:int) is **semidet**

2: Is it true that *List* has *Count* copies of *Element*?

count of(+List:list, -Element:atom, -Count:int) is **det**

3: How many of first *Element* in *List*?





























