gv Knowledge Base

```
% FILE: gv.pro
% TYPE: Prolog source
% LINE: very simple global vairable ADT
% DATE: September 25, 2015

decalre(Var,Var1) :-
  retract(binding(Var,_)),
  assert(binding(Var,Var1)).
declare(Var,Var1) :-
  assert(binding(Var,Var1)).

bind(Variable,Value) :-
  retract(binding(Variable,_)),
  assert(binding(Variable,Value)).

valueOf(Variable,Value) :-
  binding(Variable,Value).

undeclare(Var) :-
  retract(binding(Var,_)).

inc(Variable) :-
  retract(binding(Variable,Value)),
  NewValue is Value + 1,
  assert(binding(Variable,NewValue)).

dec(Variable) :-
  retract(binding(Variable,Value)),
  NewValue is Value - 1,
  assert(binding(Variable,NewValue)).

add(Variable,Number) :-
  retract(binding(Variable,Value)),
  NewValue is Value + Number,
  assert(binding(Variable,NewValue)).

displayBindings :-
```

```
  binding(Variable,Value),
  write(Variable),write(' -> '),write(Value),nl,
  fail.
displayBindings.

prepend(Variable,Value) :- % assume a list!
  retract(binding(Variable,OldValue)),
  NewValue = [Value|OldValue],
  assert(binding(Variable,NewValue)).
```