Global Variable ADT Manual

It is the manual to the Global Variable ADT. Global Variable ADT is a program designed to store variables and values in a prolog knowledgment base.

Declare - One of the most important rules, it includes a variable to the KB and associate it to a value.
Syntax: declare(Var,Val).
Example: declare(a,1).

Bind - It can change the value of especific variable already defined with 'declare'. If the user input a variable that does not exist it is going to return false.
Syntax: bind(Variable,Value).
Example: bind(a,2).

ValueOf - It can verify the value of a variable defined.
Syntax: valueOf(Variable,Value).
Example: valueOf(a,Value).
Value = 2.

Undeclare - As it says, that command can undeclare a variable.
Syntax: undeclare(Var).
Example: undeclare(a).

Increment - That rule increment a variable plus one.
Syntax: inc(Variable)
Exemple: inc(a).

Decrement - That rule decrement a variable plus one.
Syntax: dec(Variable)
Exemple: dec(a).

Add - that rule add a specific number to a variable.
Syntax: add(Variable,Number)
Exemple: % a = 2
add(a,2).
true.
% a = 4

Display Bindigns - Tha rule can list all variables and its values.
Syntax: displayBindings

Prepend - That rule assumes that the variable is a list and appends a value to its beginning.
Syntax: prepend(Variable,Value).
Exemple: prepend(a,3).
% a = [3,4]