TERMWORK 9.

problem Statement
write a prolog program to find
i) membebs of SET
ii) Concentrate two sets (concatination)
iii) delete from SET

Theory

Sets are powerful data structure that can be naturaly expressed using lists in prolog. In prolog lists element are enclosed by brackets and separated by commas.

Eg:- [1, 2, 3, 4]
[John, X]
[[mary, Joe], [bob, carol, ted, alice]]
[A, [p, h, C], 4]
[ ]

Another way to represent a list is to use the head/ tail notation (H/T)
Here the head of the list. H is represented from the tail. of the list. T. by a verticle bar The tail of a list is the original list with its first element removed. The tail of the last is always a list. even if it is empty list.

In prolog. the H/T notation is used together with unification to combine and break up lists
for example:.

Suppose we have the following list
[bob, carel. ted]
Here's the varies matche we could obtain using H/T :

[x/ y], x = bob y = [carel, ted].
[x; y/z] x = bob y = carel z = [ted]
[x, y. z/w], x = bob y = carol. z = ted w = []

* Syntax : [H/T]
```
  list ([p.q,r])
what-is ([Head|Tail]) :- list ([Head|Tail])
  ?- what-is ([Head|Tail])
   Head = p
   Tail - [q,r]
```

Program

```
list (x, [x|_]).
list (x, [_|Tail] :- list (x, Tail].

Con ([], L, L).
con ([x, | L1], L2 [X1| L3]) :- Con (L1, L2, L3)
d
del (x, [x|T], T).
del (x, [H|T], [H, T1]] :- del (x, T, T1).
```

Output
```
?- list (a, [b, a, c]).
   true.
```

```
2   ?- list(4,[2,3,4])
      true;
      false

3   ?- list(5,[0,3,6])
      false

4   ?- con([1,2,3],[4,5,6],L).
      L = [1,2,3,4,5,6]
      .

5   ?- con([1,2,3],[3,4,5],L)
      L = [1,2,3,3,4,5]

6   ?- del(1,[1,2,3],L).
        L = [2,3]

7]  ?- del(6,[4,5,6],L).
        L=[4,5]
```

10/10

24/04/23