EX.NO: 6 **DATE: 27 - 09 -2024**

INTRODUCTION TO PROLOG

AIM:

To learn PROLOG terminologies and write basic programs.

TERMINOLOGIES:

1. Atomic Terms:

   Atomic terms are usually strings made up of lower- and uppercase letters, digits, and the underscore, starting with alowercase letter.

   Ex:

   ```
   dog
   ab_c_321
   ```

2. Variables:

   Variables are strings of letters, digits, and the underscore,starting with a capital letter or an underscore.

   Ex:

   ```
   Dog
   Apple_420
   ```

3. Compound Terms:

   Compound terms are made up of a PROLOG atom and a number of arguments (PROLOG terms, i.e., atoms, numbers, variables, or other compound terms) enclosed in parentheses and separated by commas.

   Ex:

   ```
   is_bigger(elephant,X)
   f(g(X,_),7)
   ```

4. Facts:

   A fact is a predicate followed by a dot.

   Ex:

   ```
   bigger_animal(whale).
   life_is_beautiful.
   ```

5. Rules:

   A rule consists of a head (a predicate) and a body (a sequenceof predicates separated by commas).

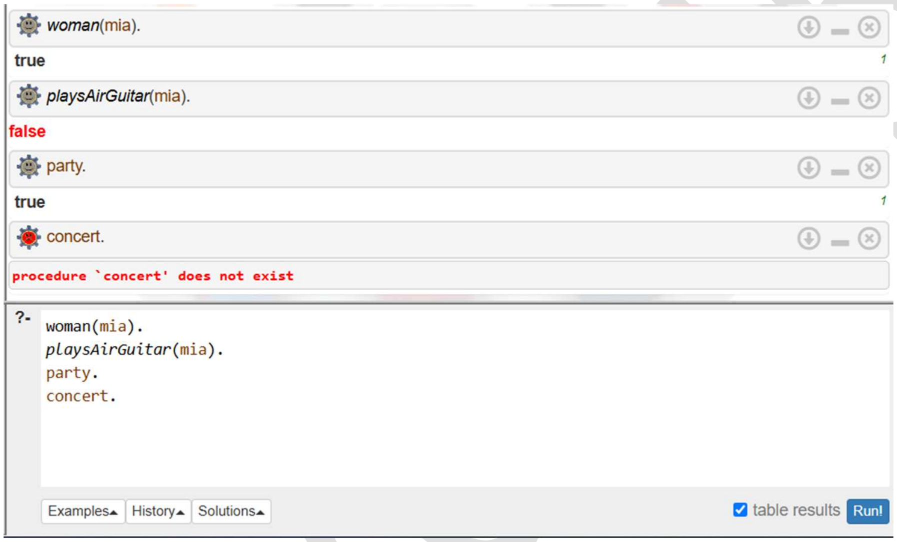
   Ex:

   ```
   is_smaller(X,Y):-is_bigger(Y,X).
   aunt(Aunt,Child):-sister(Aunt,Parent),parent(Parent,Child).
   ```

SOURCE CODE:

KB1:

woman(mia).
woman(jody).
woman(yolanda).
playsAirGuitar(jody).
party.
Query 1: ?-woman(mia).
Query 2: ?-playsAirGuitar(mia).
Query 3: ?-party.
Query 4: ?-concert.

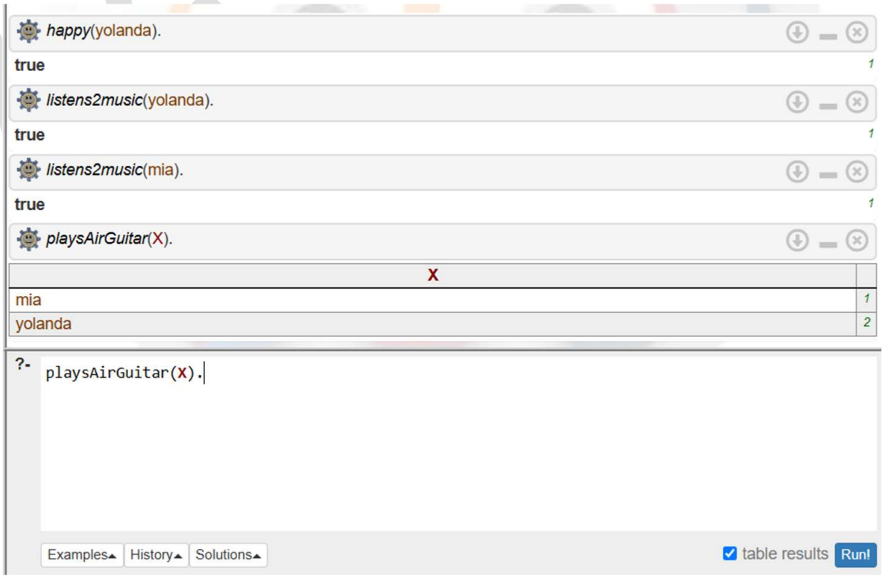


```
woman(mia).
true                                                    1
playsAirGuitar(mia).
false
party.
true                                                    1
concert.
procedure `concert' does not exist
```

```
?- woman(mia).
   playsAirGuitar(mia).
   party.
   concert.
```

KB2:

happy(yolanda).
listens2music(mia).
listens2music(yolanda):-happy(yolanda).
playsAirGuitar(mia):-listens2music(mia).
playsAirGuitar(Yolanda):-listens2music(yolanda).



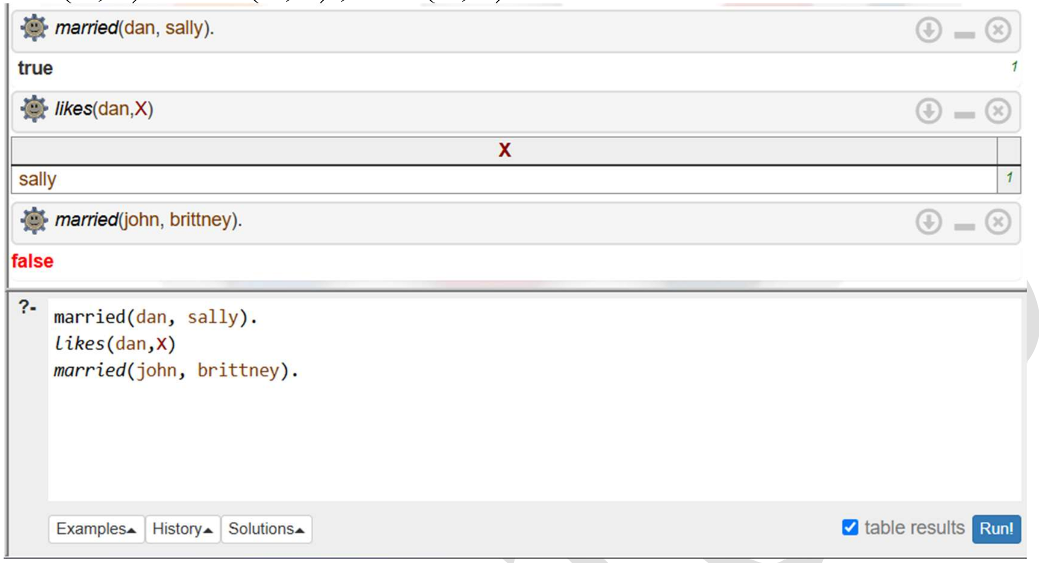
```
happy(yolanda).
true                                                    1
listens2music(yolanda).
true                                                    1
listens2music(mia).
true                                                    1
playsAirGuitar(X).
```

X	
mia	1
yolanda	2

```
?- playsAirGuitar(X).
```

KB3:

```
likes(dan,sally).
likes(sally,dan).
likes(john,brittney).
married(X,Y) :- likes(X,Y) , likes(Y,X).
friends(X,Y) :- likes(X,Y) ; likes(Y,X).
```

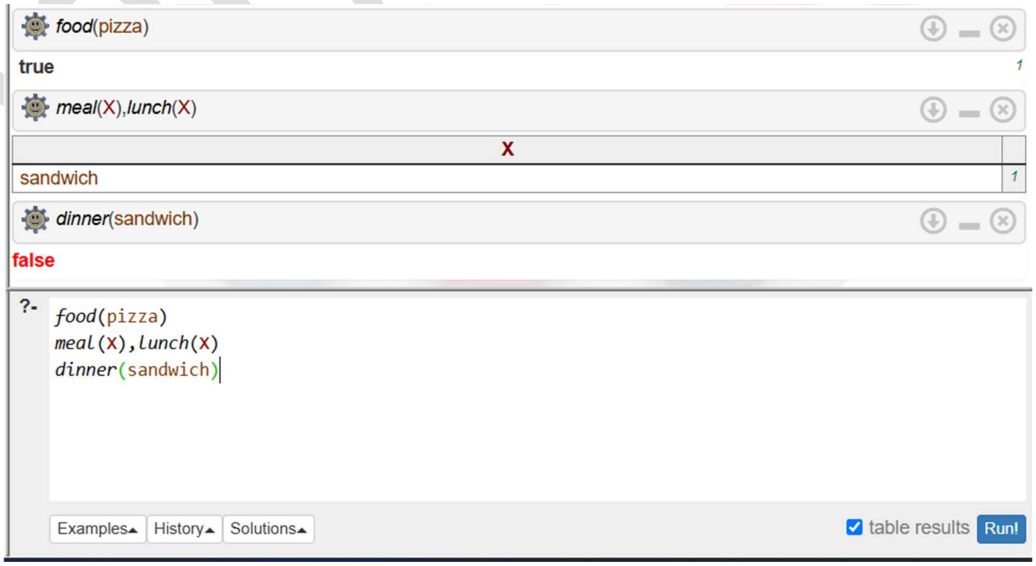


```
married(dan, sally).
true                                        1

likes(dan,X)
X
sally                                       1

married(john, brittney).
false
```

```
?- married(dan, sally).
likes(dan,X)
married(john, brittney).
```

KB4:

```
food(burger).
food(sandwich).
food(pizza).
lunch(sandwich).
dinner(pizza).
meal(X):-food(X).
```

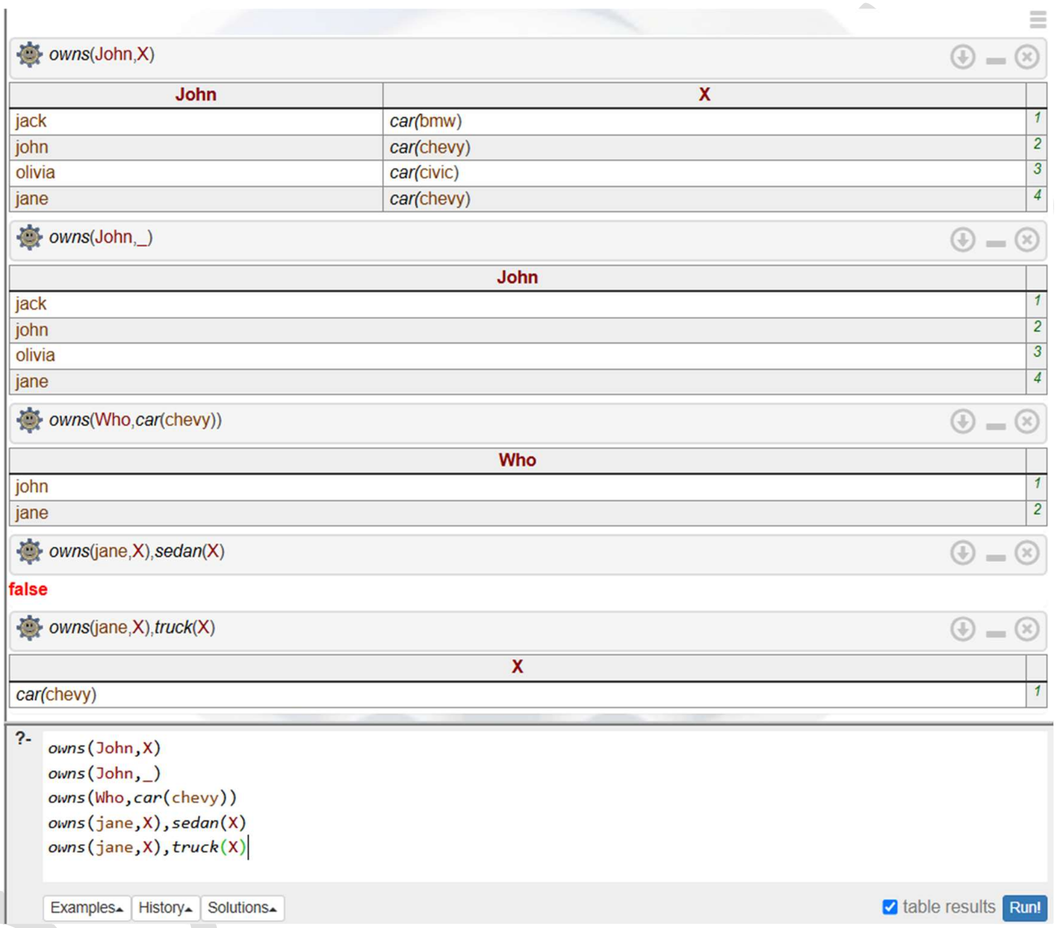


```
food(pizza)
true                                        1

meal(X),lunch(X)
X
sandwich                                    1

dinner(sandwich)
false
```

```
?- food(pizza)
meal(X),lunch(X)
dinner(sandwich)
```

KB5:

```
owns(jack,car(bmw)).
owns(john,car(chevy)).
owns(olivia,car(civic)).
owns(jane,car(chevy)).
sedan(car(bmw)).
sedan(car(civic)).
truck(car(chevy)).
```



owns(John,X)

John	X	
jack	car(bmw)	1
john	car(chevy)	2
olivia	car(civic)	3
jane	car(chevy)	4

owns(John,_)

John	
jack	1
john	2
olivia	3
jane	4

owns(Who,car(chevy))

Who	
john	1
jane	2

owns(jane,X),sedan(X)

```
false
```

owns(jane,X),truck(X)

X	
car(chevy)	1

```
?- owns(John,X)
   owns(John,_)
   owns(Who,car(chevy))
   owns(jane,X),sedan(X)
   owns(jane,X),truck(X)
```

RESULT:

Thus, we have written basic programs to learn prolog terminologies.