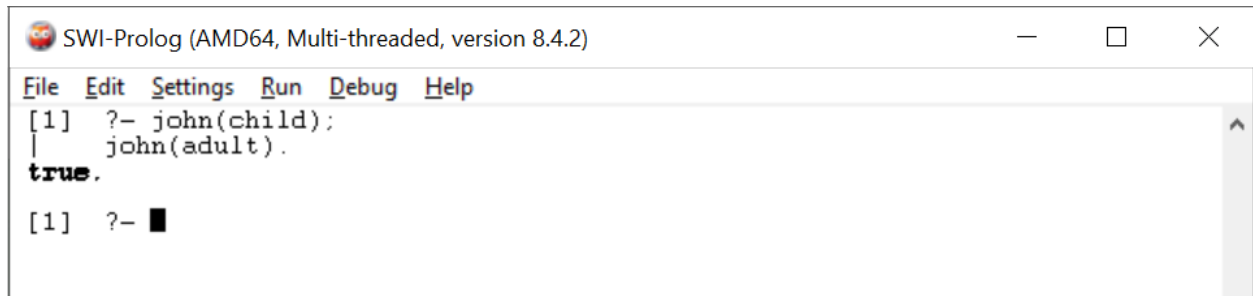


TASK3: QUESTION ANSWERING

1. Is John a child or an adult? [Adult]

Input -> john(adult).

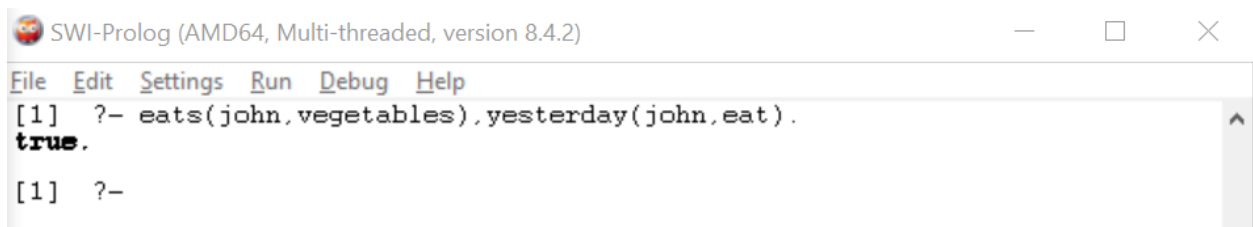
Output->



```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.2)
File Edit Settings Run Debug Help
[1] ?- john(child);
    |   john(adult).
true.
[1] ?- █
```

2. Did John eat any vegetables yesterday? [Yes]

Input -> eats(john,vegetables) and yesterday(john,eat). {NOTE: IF YOU HAVE TO USE “AND” FUNCTION THEN HAVE TO PUT “,” BETWEEN THE TWO CLAUSES.}



```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.2)
File Edit Settings Run Debug Help
[1] ?- eats(john,vegetables),yesterday(john,eat).
true.
[1] ?-
```

3. Did John buy any meat? [Yes]

Input-> buys(john,_).

It means john is buying everything in the store.

Here I write meat so that's true.

```
true.  
[1] ?- buys(john,meat).  
true
```

4. If Mary was buying a cheesecake at the same time as John, did he see her? [Yes]

Input-> johnseen(X):- buys(X,cheesecake),buys(john,cheesecake).

All the Capital letters stands for the variable

[“:-“ stands for “if” & “,” stands for “and”]

Output->

```
[1] ?- johnseen(mary).  
true.
```

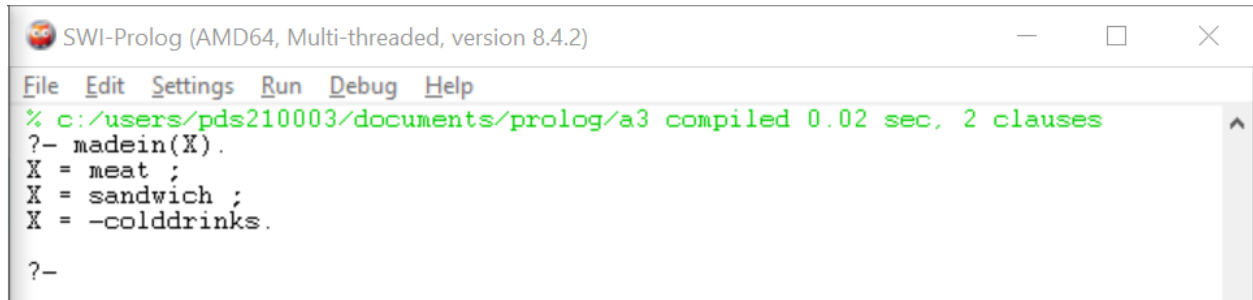
5. Are onions made in Jason's Deli? [No]

Input ->

If we put variable in the argument X, then it will shows you the all the possible outcomes.

Suppose here I have take a input as (madein(X).) so it will shows that what is made in the deli of Jason.

Output->



```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.2)
File Edit Settings Run Debug Help
% c:/users/pds210003/documents/prolog/a3 compiled 0.02 sec, 2 clauses
?- madein(X).
X = meat ;
X = sandwich ;
X = -colddrinks.
?-
```

```
[1] ?- madein(onions).
false.
```

6. What is John going to do with the Beefeater? [Eat it]

Input-> beefeater(john, X);

This clause will show all the possible outcomes for the john did with beefeater.

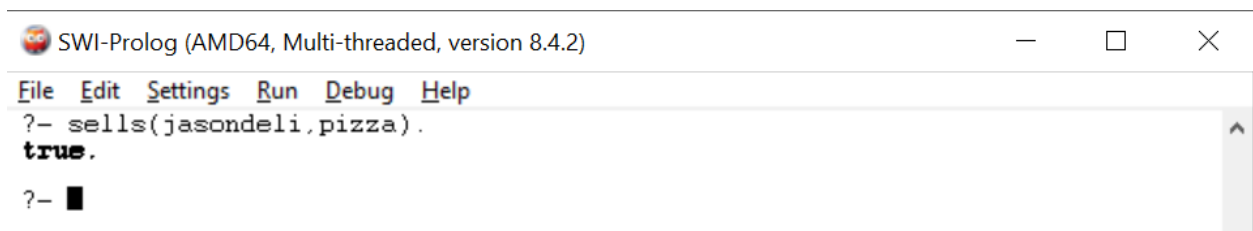
Output:-

```
?- beefeater(john,X).
X = eatit.
```

7. Does Jason's Deli sell pizza? [Yes]

Input-> sells(jasondeli,pizza).

Output:



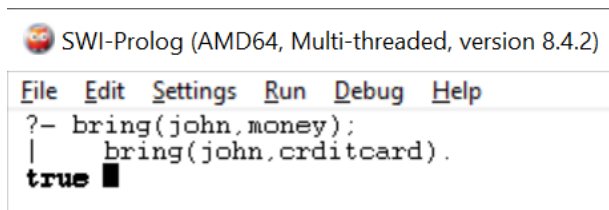
```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.2)
File Edit Settings Run Debug Help
?- sells(jasondeli,pizza).
true.
?-
```

8. Did John bring money or a credit card to the deli? [Yes]

Input:- {In prolog “;” means “OR”.}

So as a input bring(john,money) or bring(john,creditcard) gives you answer TRUE.

Output:-

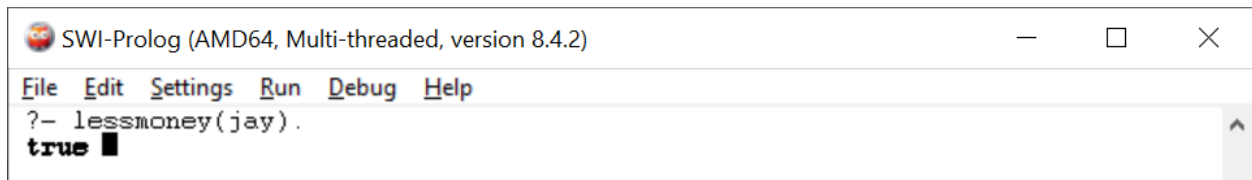


```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.2)
File Edit Settings Run Debug Help
?- bring(john,money);
   bring(john,creditcard).
true
```

9. Did John have less money after going to the deli? [Yes]

Input-> lessmoney(X) :- buys(_,cheesecake),bring(_money).

Output:-



```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.2)
File Edit Settings Run Debug Help
?- lessmoney(jay).
true
```

10. Are there other people at Jason's deli while John is there? [Yes - there is staff!]

Input-> "issaff(john). "Gives me true value.

Any other from john gives me false value.

Output:-

```
?- isstaff(john).
true.
?- isstaff(jay).
false.
?-
```

11. Is John a vegetarian? [No]

Input clause : vegetarian(-john)

Output:-

```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.2)
File Edit Settings Run Debug Help
?- vegetarian(john).
false.
?- vegetarian(jay).
false.
?- █
```

12. Who owns cheese slices at the deli? [Jason's Deli]

Input clause-> owns(cheesesclices,jasondeli).

Output:-

```
?- owns(cheesesclices,jasondeli).
false.
?- owns(cheeseslices,jasondeli).
true.
?- █
```

13. Did John have an ounce of roast beef? [Yes]

Input clause : have(_roastbeef,ounce.). ["_" stands for "all"]

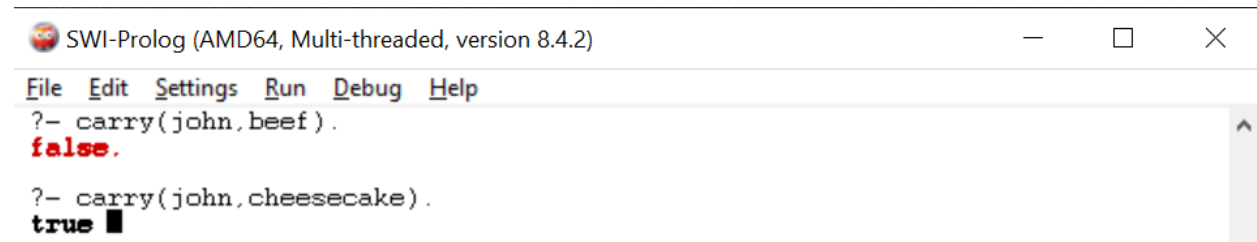
```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.2)
File Edit Settings Run Debug Help
?- have(john,roastbeef,ounce).
true.
?- have(rakesh,roastbeef,ounce).
true.
?- █
```

14. Could John carry the food he purchased? [Yes]

Input clause:

Carry(john,X) :- buys(X,cheesecake),bring(john,money).

Output:-



```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.2)
File Edit Settings Run Debug Help
?- carry(john,beef).
false.
?- carry(john,cheesecake).
true ■
```