Example -1 : Food

Let's start with a simple example to understand the terminologies. We will provide Facts and Rules to the prolog system and then we will ask queries and we will see what prolog interpreter returns as an answer and why.

Facts	**English meanings of Facts, Rules & Goals**
food(burger).	// burger is a food
food(sandwich).	// sandwich is a food
food(pizza).	// pizza is a food
lunch(sandwich).	// sandwich is a lunch
dinner(pizza).	// pizza is a dinner
Rules	
meal(X) :- food(X).	// Every food is a meal OR Anything is a meal if it is a food
Queries / Goals & answers	
?- food(pizza). true.	// Is pizza a food? Answer : true Explanation : Here prolog will return 'true or yes'. Because first, prolog interpreter will trace through the facts and rules in top-down manner and when it can find the match it will provide the answer and in this case it can find the exact match.
?- meal(X), lunch(X). X = sandwich.	// Which food is meal and lunch? OR What is both meal and lunch? Answer : X = sandwich. Explanation : Here in this query we have provided two subgoals where "," comma means 'and'. Prolog always tries to satisfy subgoals in left-to-right manner, so first try to get left most goal i.e. meal(X). But meal(X) rule says - X is a meal if X is a food. So, now we will look for food(X). Here X is a variable and it can bound with any related value. So, in food(X) - X can be burger, sandwich or pizza as per our facts. But second goal says that X should also be lunch. Now we look for X value in lunch(X) and i.e. sandwich. So now we find which food(X) values matches with lunch(X) and the answer is

	sandwich. You can learn more about these kind of search process queries in Conjunction & backtracking.
?- dinner(sandwich). false.	// Is sandwich a dinner? Answer : false. Explanation : In this case prolog will find the 'dinner' predicate and will match the argument inside the bracket. But it will return 'false or no' since it cannot find the match.