

1. Write a prolog program that store countries with their federal language used in a list form using the

predicate lang_country. Define the right rule to for the following:

To display countries that uses some specific language such as English or Amharic as their official or federal language. It should count the number of countries that use the language in addition to the list.

To display the language used in a country taking the name of the country as input.

Answer

% Facts: countries and their languages

lang_country(usa, english).

lang_country(uk, english).

lang_country(ethiopia, amharic).

lang_country(kenya, english).

lang_country(sudan, arabic).

lang_country(france, french).

lang_country(italy, italian).

lang_country(japan, japanese).

lang_country(germany, german).

lang_country(spain, spanish).

% Rule 1: Find countries by language

countries_with_language(Language) :-

findall(Country, lang_country(Country, Language), Countries),

write('Countries that use '), write(Language), write(' as their official language:'), nl,

write_list(Countries),

length(Countries, Count),

write('Number of countries: '), write(Count), nl.

% Rule 2: Find language by country

language_of_country(Country) :-

lang_country(Country, Language),

write('The official language of '), write(Country), write(' is '), write(Language), nl.

language_of_country(Country) :-

\+ lang_country(Country, _),

write('No data available for the country: '), write(Country), nl.

% Helper Print list

write_list([]).

write_list([H|T]) :-

write(' '), write(H), nl,

write_list(T).

2. Create three predicates in a knowledge base that add list element at the beginning, at the middle and at the end of a list.

Answer

% 1. Predicate to add an element at the beginning of a list

add_at_beginning(Element, List, [Element|List]).

% 2. Predicate to add an element in the middle of a list

add_at_middle(Element, List, Result) :-

length(List, Length),

```

MiddleIndex is Length // 2,      % Find the middle position
split_at(List, MiddleIndex, Front, Back), % Split the list at the middle
append(Front, [Element|Back], Result).

```

% 3. Predicate to add an element at the end of a list

```

add_at_end(Element, List, Result) :-
    append(List, [Element], Result).

```

% Helper predicate to split a list at a specific index

```

split_at(List, 0, [], List).
split_at([H|T], N, [H|Front], Back) :-
    N > 0,
    N1 is N - 1,
    split_at(T, N1, Front, Back).

```

3. List any five built in predicates used in prolog related to list. Use example to explain each.

1. findall/3

- **Purpose:** Finds all solutions to a query and stores them in a list.
- **Example:** lang_country(usa, english).
lang_country(uk, english).
lang_country(ethiopia, amharic).
?- findall(Country, lang_country(Country, english), Countries).
- **Output:** Countries = [usa, uk].

2. last/2

- **Purpose:** Retrieves the last element of a list.
- **Example:** ?- last([a, b, c, d], Last).
- **Output:** Last = d.

3. max_list/2

- **Purpose:** Finds the maximum value in a list of numbers.
- **Example:** ?- max_list([1, 5, 3, 9, 2], Max).
- **Output:** Max = 9.

4. sum_list/2

- **Purpose:** Calculates the sum of all elements in a list of numbers.
- **Example:** ?- sum_list([1, 2, 3, 4, 5], Sum).
- **Output:** Sum = 15.

5. msort/2

- **Purpose:** Sorts a list in ascending order without removing duplicates.
- **Example:** ?- msort([3, 1, 2, 3, 4, 2], Sorted).
- **Output:** Sorted = [1, 2, 2, 3, 3, 4]

4. A prolog program that define the following predicate

- Cities and countries / 1
Cities in Africa
Example Addis Ababa, Ethiopia.
- City Country/2
A rule that displays list of Cities for a country name given by user.
- Add City with three arguments this should check if that City country pair exist and if the pair doesn't exist it should add it to the list.

Answer

% 1. Define the initial list of city-country pairs

```
cities_in_africa([
    city('Addis Ababa', 'Ethiopia'),
    city('Nairobi', 'Kenya'),
    city('Cairo', 'Egypt'),
    city('Lagos', 'Nigeria'),
    city('Cape Town', 'South Africa'),
    city('Kampala', 'Uganda'),
    city('Algiers', 'Algeria'),
    city('Accra', 'Ghana'),
    city('Tunis', 'Tunisia'),
    city('Dar es Salaam', 'Tanzania')
]).
```

% 2. cities_and_countries/1

% Displays all cities and their respective countries in the list

cities_and_countries(List) :-

```
    cities_in_africa(List),
    write('Cities in Africa:'), nl,
    print_cities(List).
```

% Helper predicate to print cities and countries

print_cities([]).

print_cities([city(City, Country) | Rest]) :-

```
    write(City), write(' '), write(Country), nl,
    print_cities(Rest).
```

% 3. city_country/2

% Displays the list of cities for a given country

city_country(Country, Cities) :-

```
    cities_in_africa(List),
    findall(City, member(city(City, Country), List), Cities),
    ( Cities = [] ->
        write('No cities found for '), write(Country), nl
    ; write('Cities in '), write(Country), write(' are: '), write(Cities), nl
    ).
```

% 4. add_city/3

% Adds a new city-country pair if it doesn't already exist

add_city(City, Country, ResultList) :-

```
    cities_in_africa(List),
```

```
( member(city(City, Country), List) ->  
  write('The city-country pair already exists. '), nl,  
  ResultList = List  
; append(List, [city(City, Country)], ResultList),  
  write('City-country pair added successfully. '), nl  
).
```