## Indian Institute of Information Technology Surat

****

# Lab Report on

# Artificial Intelligence (CS 701) Practical

**Submitted by**

### [RAHUL KUMAR SINGH] (UI21CS44)

**Course Faculty**

### Dr. Ritesh Kumar

### Mrs. Archana Balmik

## Department of Computer Science and Engineering

## Indian Institute of Information Technology Surat

## Gujarat-394190, India

**Aug-2024**

## Lab No: 2

## Aim:

To develop a Prolog-based family tree system to query relationships, ages, and family structure for a comprehensive genealogy analysis

## Description:

Use Prolog for answering the following questions (load the rules in the file familytree.pl):

1. Is Albert a parent of Peter?

2. Who is the child of Jim?

3. Who are the parents of Brian?

4. Is Irene a grandparent of Brian?

5. Find all the grandchildren of Irene

6. Now add the following rule to familytree.pl and re-consult:

older(Person1, Person2) :-

yearOfBirth(Person1, Year1),

yearOfBirth(Person2, Year2),

Year2 &gt; Year1.

7. Who is older than Pat?

8. Who is younger than Darren?

9. List the siblings of Sandra.

10. Who is the older brother of Sandra?

11. Find the predecessors of Kyle.

12. Does Kate have a sister?

13. Is Albert a parent?

14. Now add the following rule to familytree.pl and re-consult:

male1(person(&#39;Barry&#39;, &#39;Drake&#39;)).

male1(person(&#39;Jim&#39;, &#39;Fried&#39;)).

female1(person(&#39;Dot&#39;, &#39;Kanga&#39;)).

15. Find the last names of all the males and females.

## Code:

**Program**

parent(albert, jim).

parent(albert, peter).

parent(jim, brian).

parent(john, darren).

parent(peter, lee).

parent(peter, sandra).

parent(peter, james).

parent(peter, kate).

parent(petr, kyle).

parent(brian, jenny).

parent(irene, jim).

parent(irene, peter).

parent(pat, brian).

parent(pat, darren).

parent(amanda, jenny).

female(irene).

female(pat).

female(lee).

female(sandra).

female(jenny).

female(amanda).

female(kate).

male(albert).

male(jim).

male(peter).

male(brian).

male(john).

male(darren).

male(james).

male(kyle).

yearOfBirth(irene, 1923).

yearOfBirth(pat, 1954).

yearOfBirth(lee, 1970).

yearOfBirth(sandra, 1973).

yearOfBirth(jenny, 2004).

yearOfBirth(amanda, 1979).

yearOfBirth(albert, 1926).

yearOfBirth(jim, 1949).

yearOfBirth(peter, 1945).

yearOfBirth(brian, 1974).

yearOfBirth(john, 1955).

yearOfBirth(darren, 1976).

yearOfBirth(james, 1969).

yearOfBirth(kate, 1975).

yearOfBirth(kyle, 1976).

older(Person1, Person2) :-

yearOfBirth(Person1, Year1),

yearOfBirth(Person2, Year2),

Year2 > Year1.

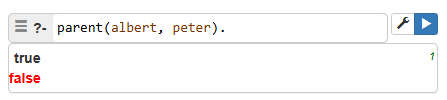
male1(person('Barry', 'Drake')).

male1(person('Jim', 'Fried')).

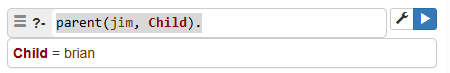
female1(person('Dot', 'Kanga')).

## Output:

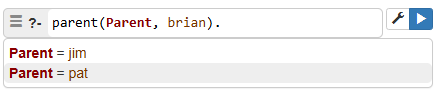
**Q1**

****

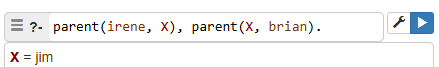
**Q2**

****

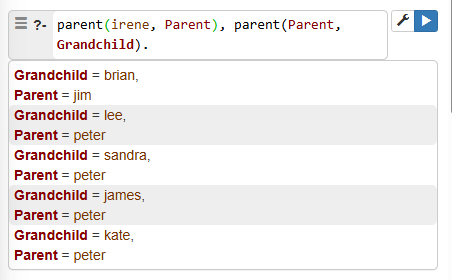
**Q3**

****

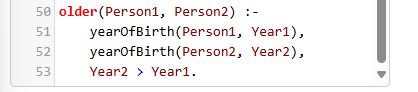
**Q4**

****

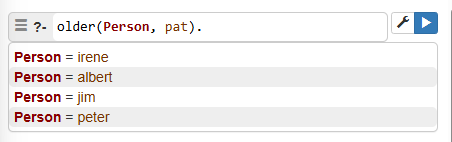
**Q5**

****

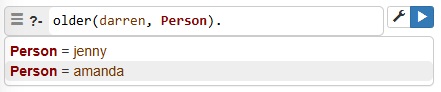
**Q6**

****

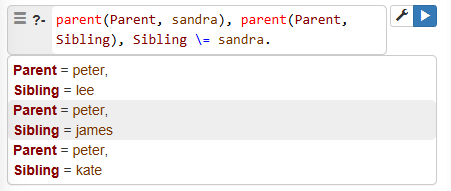
**Q7**

****

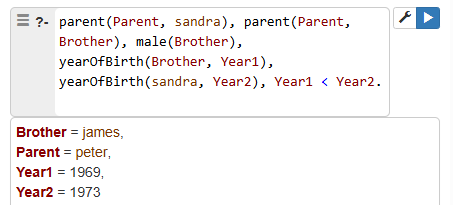
**Q8**

****

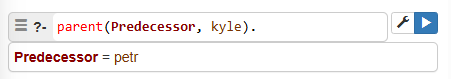
**Q9**

****

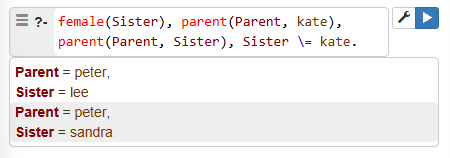
**Q10**

****

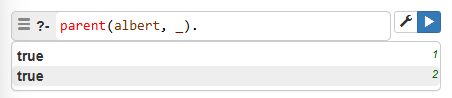
**Q11**

****

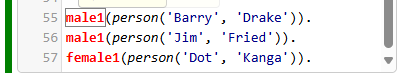
**Q12**

****

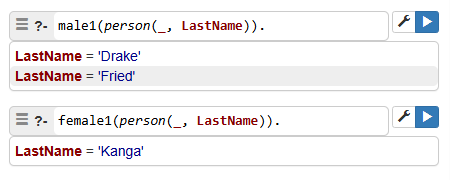
**Q13**

****

**Q14**

****

**Q15**

****

## Conclusion:

* The Prolog system shows detailed family connections and structures.
* Queries help identify relationships and compare ages easily.
* Updating rules makes the data more accurate.
* The system helps with detailed family research and records.