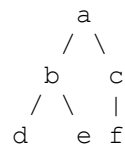


**Ex. No. 5****Logic Programming****1. Implement using pyDatalog:**

Assume given a set of facts of the form `father(name1,name2)` (`name1` is the father of `name2`), .

- a. Define a predicate `brother(X,Y)` which holds iff `x` and `y` are brothers.
- b. Define a predicate `cousin(X,Y)` which holds iff `x` and `y` are cousins.
- c. Define a predicate `grandson(X,Y)` which holds iff `x` is a grandson of `y`.
- d. Define a predicate `descendent(X,Y)` which holds iff `x` is a descendent of `y`.
- e. Consider the following genealogical tree:



What are the answers generated by your definitions for the queries:

```
brother(X,Y)
cousin(X,Y)
grandson(X,Y)
descendent(X,Y)
```

**2. Encode the following facts and rules in pyDatalog:**

- Bear is big
- Elephant is big
- Cat is small
- Bear is brown
- Cat is black
- Elephant is gray
- An animal is dark if it is black
- An animal is dark if it is brown

Write a query to find which animal is dark and big.

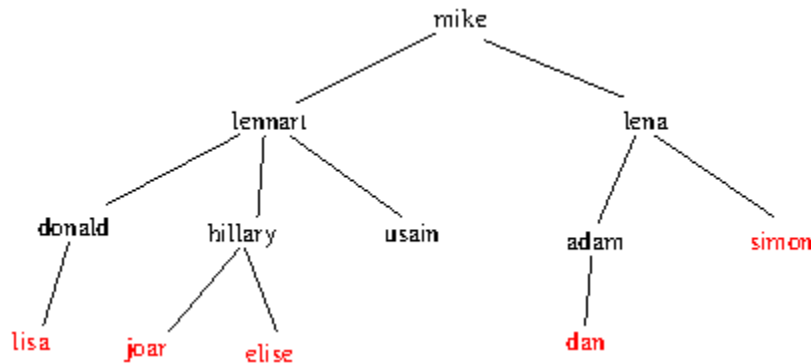
3. The following are the marks scored by 5 students.

Student Name	Mark
Ram	90
Raju	45
Priya	85
Carol	70
Shyam	80

Enter the above data using pyDatalog.

Write queries for the following:

- Print Student name and mark of all students.
  - Who has scored 80 marks?
  - What mark has been scored by Priya?
  - Write a rule 'passm' denoting that pass mark is greater than 50. Use the rule to print all students who failed.
  - Write rules for finding grade letters for a marks and use the rule to find the grade letter of a given mark.
4. Solve the set of queries in the previous question using imperative programming paradigm in Python. Store the data in a dictionary.
5. Write a recursive program to find factorial of a number using pyDatalog.
6. Implement the following using pyDatalog for the family tree where hillary lena, lisa, and elise are female



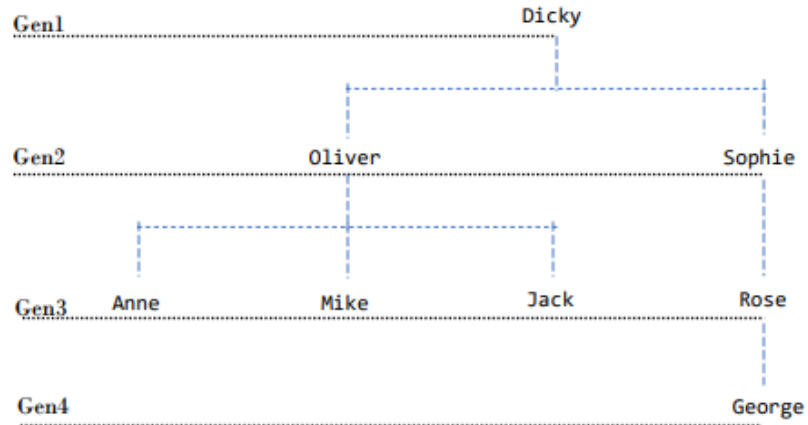
Define new predicates (in terms of rules using male/1, female/1 and parent/2) for the following family relations:

- |            |             |                 |            |                 |
|------------|-------------|-----------------|------------|-----------------|
| (a) father | (b) sister  | (c) grandmother | (d) cousin | (e) grandfather |
| (f) mother | (g) brother | (h) uncle       | (i) aunty  |                 |

Write a query to return the cousin of adam

Write a query to return the grandfather of elise

7. Implement the following using pyDatalog for the family tree



Males: Dicky, Oliver, Mike, Jack, George

Females: Anne, Rose, Sophia

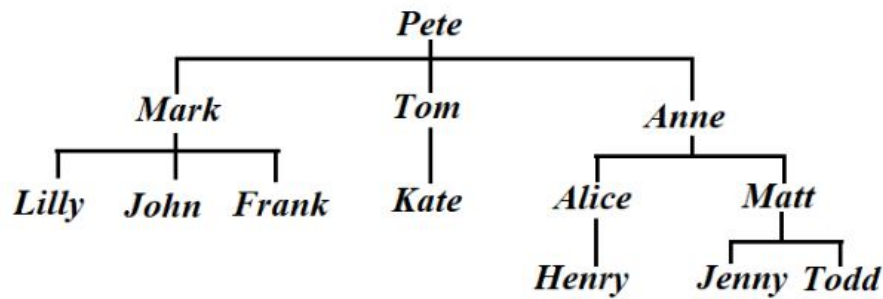
Define new predicates for the following family relations: Father, Mother, Sister, Brother, Grandmother, Grandfather, Ancestor, Cousin, Uncle, Son and Daughter.

Your program should be able to answer the following question.

- Was George the parent of Oliver?
- Who was Oliver's parent?
- Who were the children of Oliver?
- Who were the brothers of Anne?
- Who were the cousins of Rose?

Create any of your own questions (five) and test the answers.

8. Implement the following using pyDatalog for the family tree



Create a simple database containing facts and inference rules. Include facts about family members, such as *male*, *female*, *parent*. Then add rules such as *sister*, *brother*, *sibling*, *father*, *mother*, *grandparent*. Run queries that will answer the following questions about family members: (a) Is Pete Mark's parent? (b) Is Anne Jenny's parent? (c) Who is Todd's father? (d) Who is Tom's sibling? (e) Who is Lilly's brother? (f) Who is Henry's grandparent? (g) Who is Alice's sister? (h) Is Frank Kate's brother? (i) Who is Matt's mother? (j) Is Mark Anne's brother?

Submission file should include your database and screenshots of the results of all queries.