WEEK - 9

**Name- Aditya Mishra**

**Reg. No. – RA2111003011817**

**Section- M2**

**QUESTION 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: a

/ \

b c

/ \ |

d e f

What are the answers generated by your definitions for the queries: brother(X,Y)

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

**CODE:**

from pyDatalog import pyDatalog pyDatalog.create\_terms('a,b,c,d,e,f,brother,cousin,grandson,descendent,X,Y')

+brother('b','c')

+brother('d','e')

+cousin('d','f')

+cousin('e','f')

+grandson('d','a')

+grandson('e','a')

+grandson('f','a')

+descendent('b','a')

+descendent('c','a')

+descendent('d','b')

+descendent('f','c') print(pyDatalog.ask('brother(X,Y)')) print(pyDatalog.ask('cousin(X,Y)')) print(pyDatalog.ask('grandson(X,Y)')) print(pyDatalog.ask('descendent(X,Y)'))

**Output-**

{('d', 'e'), ('b', 'c')}

{('d', 'f'), ('e', 'f')}

{('e', 'a'), ('d', 'a'), ('f', 'a')}

{('f', 'c'), ('b', 'a'), ('c', 'a'), ('d', 'b')}

**QUESTION 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:

a. Print Student name and mark of all students.

b. Who has scored 80 marks?

c. What mark has been scored by Priya?

d. Write a rule ‘passm’ denoting that pass mark is greater than 50. Use the rule to print all students who failed.

e. Write rules for finding grade letters for a marks and use the rule to find the grade letter of a given mark.

**CODE:**

from pyDatalog import pyDatalog pyDatalog.create\_terms('X,Y,Z,student,marks,passm,grades')

+student('ram')

+student('raju')

+student('priya')

+student('carol')

+student('shyam')

+marks('90','ram')

+marks('45','raju')

+marks('85','priya')

+marks('70','carol')

+marks('80','shyam')

+grades('ram','O')

+grades('priya','A')

+grades('shyam','A')

+grades('carol','B')

+grades('raju','E') print(marks(X,Y)) print(marks('80',X))

print(marks(X,'priya'))

passm(X)<=grades(X,'E') print(passm(X))

**OUTPUT:**

X | Y

….|-…….

80 | shyam

70 | carol

85 | priya

45 | raju

90 | ram

X

- shyam

X

--

85

X

raju