

## **EX.NO : 6**

### **PROLOG**

#### **AIM :**

To develop a family tree program using PROLOG with all possible facts , rules and queries.

#### **SOURCE CODE:**

##### **KNOWLEDGE BASE:**

```
/*FACTS :: */
```

```
male(peter).
```

```
male(john). male(chris).
```

```
male(kevin).
```

```
female(betty).
```

```
female(jeny). female(lisa).
```

```
female(helen).
```

```
parentOf(chris,peter).
```

```
parentOf(chris,betty).
```

```
parentOf(helen,peter).
```

```
parentOf(helen,betty).
```

```
parentOf(kevin,chris).
```

```
parentOf(kevin,lisa).
```

```
parentOf(jeny,john).
```

```
parentOf(jeny,helen).
```

```
/*RULES :: */
```

```
/* son,parent
```

```
* son,grandparent*/
```

```
father(X,Y):- male(Y), parentOf(X,Y).
```

```
mother(X,Y):- female(Y), parentOf(X,Y).
```

```
grandfather(X,Y):- male(Y),
```

```
parentOf(X,Z), parentOf(Z,Y).
```

grandmother(X,Y):- female(Y),  
parentOf(X,Z), parentOf(Z,Y).

brother(X,Y):- male(Y),  
father(X,Z),  
father(Y,W), Z==W.

sister(X,Y):- female(Y),  
father(X,Z),  
father(Y,W), Z==W.

## OUTPUT :

male(peter)	1	2
true		
father(chris,peter)	1	2
true		
father(chris,betty)	1	2
false		
grandfather(kevin,peter)	1	2
true		
grandfather(jerry,peter)	1	2
true		
grandmother(jerry,peter)	1	2
false		
mother(chris,X)	1	2
X = betty		
brother(helen,chris)	1	2
true		
brother(chris,helen)	1	2
false		
father(X,Y)	1	2
X = chris, Y = peter X = helen, Y = peter X = jerry, Y = john X = kevin, Y = chris		
mother(X,Y)	1	2
X = chris, Y = betty X = helen, Y = betty X = kevin, Y = lisa X = jerry, Y = helen		

```
grandmother(X,Y)
X = kevin,
Y = betty
X = jerry,
Y = betty

grandfather(X,Y)
X = kevin,
Y = peter
X = jerry,
Y = peter
```

```
brother(X,Y)
X = Y, Y = chris
X = helen,
Y = chris
X = Y, Y = kevin

sister(X,Y)
X = Y, Y = jerry
X = chris,
Y = helen
X = Y, Y = helen
```

**RESULT :** Thus the above python code is executed successfully and output is verified.