# Exercises:

## 1.1 Which of the following sequences of characters are atoms, which are variables, and which are neither?

1. vINCENT – atom
2. Footmassage – variable
3. variable23 – atom
4. Variable2000 – variable
5. big\_kahuna\_burger – atom
6. 'big kahuna burger' – atom
7. big kahuna burger – neither
8. 'Jules' – atom
9. \_Jules – variable
10. '\_Jules' - atom

## 1.2 Which of the following sequences of characters are atoms, which are variables, which are complex terms, and which are not terms at all? Give the functor and arity of each complex term.

1. loves(Vincent,mia) - complex term, functor: loves, arity: 2
2. 'loves(Vincent,mia)' - atom
3. Butch(boxer) - not a term
4. boxer(Butch) - complex term, functor: boxer, arity: 1
5. and(big(burger),kahuna(burger)) - complex term, functor: and, arity: 2
6. and(big(X),kahuna(X)) - complex term, functor: and, arity: 2
7. \_and(big(X),kahuna(X)) - not a term
8. (Butch kills Vincent) - not a term
9. kills(Butch Vincent) - not a term
10. kills(Butch,Vincent - not a term

## 1.3 How many facts, rules, clauses, and predicates are there in the following knowledge base? What are the heads of the rules, and what are the goals they contain?

woman(vincent).

woman(mia).

man(jules).

person(X) :- man(X); woman(X).

loves(X,Y) :- father(X,Y).

father(Y,Z) :- man(Y), son(Z,Y).

father(Y,Z) :- man(Y), daughter(Z,Y).

number of facts: 3

number of rules: 4

number of clauses: 7

number of predicates: 7

|  |  |
| --- | --- |
| Head | Goals |
| Person(X) | man(X), woman(X) |
| loves(X,Y) | father(X,Y) |
| father(Y, Z) | man(Y), son(Z, Y) |
| father(Y, Z) | man(Y), daughter(Z, Y) |

## 1.4 Represent the following in Prolog:

1. Butch is a killer.

Ans：killer(butch).

1. Mia and Marcellus are married.

Ans：married(mia, marcellus).

1. Zed is dead.

Ans: dead(zed).

4. Marcellus kills everyone who gives Mia a footmassage.

Ans: kills(marcellus, X) :- givesFootMassage(X, mia).

5. Mia loves everyone who is a good dancer.

Ans: loves(mia, X) :- goodDancer(X).

6. Jules eats anything that is nutritious or tasty.

Ans: eats(jules, X) :- nutritious(X).

eats(jules, X) :- tasty(X).

## 1.5 Suppose we are working with the following knowledge base:

wizard(ron).

hasWand(harry).

quidditchPlayer(harry).

wizard(X) :- hasBroom(X), hasWand(X).

hasBroom(X) :- quidditchPlayer(X).

## How does Prolog respond to the following queries?

1. wizard(ron). -> true

2. witch(ron). -> undefined procedure

3. wizard(hermione). -> false

4. witch(hermione). -> undefined procedure

5. wizard(harry). -> true

6. wizard(Y). -> Y = ron ; Y = harry.

7.witch(Y). -> undefined procedure