# Term unification

Two terms unify

- if they are the same term or
- if they contain variables that can be uniformly instantiated with terms in such a way that the resulting terms are equal.

Write how will prolog answer this query:

?- X = 1, X = 2.

false

?- 2 = 1 + 1.

false

?- X = Y.

X = Y or true or yes

?-  loves(X,_)  =  loves(martin,kate).

X = martin

?-  loves(X,X)  =  loves(martin,kate).

false

Highlight parts of the structure to which the variables X and Y unify:

```
?-  k(s(g),Y)  =  k(X,t(k)).

   X  =  s(g)
   Y  =  t(k).
```

```
?- g(f(a,b), X) = g(f(Z,b), Z).

   X = a
   Z = a.
```

```
?- personal_data(A, A) = personal_data(B, name(B)).

   A = B, B = name(B).
```

# Which of the following pairs of terms unify

▣bread  =  bread

❑’Bread’  =  bread

▣’bread’  =  bread

▣Bread  =  bread

❑bread  =  sausage

❑food(bread)  =  bread

▣food(bread)  =  X

▣food(X)  =  food(bread)

▣food(bread,X)  =  food(Y,sausage)

❑food(bread,X,beer)  =  food(Y,sausage,X)

❑food(bread,X,beer)  =  food(Y,kahuna_burger)

❑food(X)  =  X

▣meal(food(bread),drink(beer))  =  meal(X,Y)

Define a predicate child/2

For all X and Y, Y is a child of X, if X is parent of Y.

```
child(Y, X) :- parent(X, Y).
```

Define a predicate has_child/1

Exist X such as X is a parent.

```
has_child(X) :- parent(X, _).
```

Define a predicate mother/2

```
mother(M,C) :- woman(M), parent(M,C).
```

Fill in the missing part ___to define a predicate sister/2

sister(S1,S2) :- parent(P, S1), parent(P, S2), ___________.

```
woman(S1)
```

Fill in the missing part ___to define a predicate siblings/2

siblings(S1, S2) :- ______________, S1 \= S2.

```
parent(P, S1), parent(P, S2)
```