Knowledge Representation in Prolog involves encoding facts, rules, and relationships in a way that allows logical reasoning. Prolog (Programming in Logic) is a **declarative programming language** commonly used in **Artificial Intelligence (AI)** and **Expert Systems** for this purpose.

🔹 Basic Concepts of Knowledge Representation in Prolog

1. Facts

- Represent simple assertions about the world.
- Syntax: fact_name(arguments).

Example:

parent(john, mary).

male(john).

female(mary).

Meaning:

- John is a parent of Mary.
- John is male.
- Mary is female.

2. Rules

- Express relationships based on conditions (logical implications).
- Syntax: head :- body.
- Read as: *head is true if body is true*.

Example:

father(X, Y) :- parent(X, Y), male(X).

Meaning:

- X is the father of Y if X is a parent of Y and X is male.

3. Queries

- Ask questions to Prolog about the facts and rules.
- Syntax: ?- question.

Example:

?- father(john, mary).

Returns:

true.

4. Variables

- Start with a capital letter (e.g., X, Y) and are used in rules and queries.

Example:

?- parent(X, mary).

Asks: Who is a parent of Mary?

🔹 **Knowledge Base Example**

% Facts

parent(john, mary).

parent(mary, susan).

parent(john, paul).

male(john).

female(mary).

female(susan).

% Rule

grandparent(X, Y) :- parent(X, Z), parent(Z, Y).

Query:

?- grandparent(john, susan).

Answer:

true.

🔹 **Why Use Prolog for Knowledge Representation?**

- Easy to model logical relationships.
- Supports **backtracking** to find all solutions.
- Powerful for expert systems, natural language processing, and problem-solving tasks.

If you'd like help writing a specific knowledge base or solving a problem in Prolog, just let me know!