5 paradigms of programming explain

1. Imperative Programming

Definition: This paradigm focuses on giving the computer a sequence of commands to perform.

How it works: You write code that changes the program's state using statements like assignments and loops.

Languages: C, Python (partially), Java.

2. Object-Oriented Programming (OOP)

Definition: Based on "objects" that contain data (attributes) and behavior (methods).

Core Concepts: Encapsulation, inheritance, polymorphism, abstraction.

Languages: Java, C++, Python, C#.

3. Functional Programming

Definition: Treats computation as the evaluation of mathematical functions and avoids changing state or mutable data.

Features: Pure functions, immutability, recursion, higher-order functions.

Languages: Haskell, Lisp, Scala, Python (partially).

4. Logic Programming

Definition: Uses formal logic to express facts and rules about problems within a system.

Execution: The program answers queries based on the rules and facts defined.

Languages: Prolog.

5. Declarative Programming

Definition: Focuses on what should be done, not how to do it.

How it differs: Unlike imperative, it doesn’t describe step-by-step control flow.

Languages: SQL, HTML, CSS.

Example (SQL):