**Class and object**

**Class:**

**A class in Java is like a blueprint or template for creating objects. It defines properties (attributes) and behaviors (methods) that the objects created from the class will have.**

**Example: Think of a class like the blueprint for a table. It specifies the dimensions, material, and design. However, the blueprint itself is not a table; it's just a plan for creating tables.**

**In programming terms, you are the class that gives the blueprint (instructions) to a carpenter (the program). The carpenter follows these instructions to create a real table (object).**

**Object:**

**An object is an instance of a class. It is a real-world entity created based on the blueprint (class). In Java, everything revolves around objects. Once the JVM creates an object using the class as a blueprint, the object will have the properties and behaviors defined by the class.**

**Example: If the blueprint is the class, then the actual table made by the carpenter is the object. Each table (object) may have its own state (specific size, color, etc.) but follows the general structure defined by the blueprint (class).**