What types of problems Design pattern solves

Design patterns are like tried-and-true recipes for solving common problems in software development. They help us create well-organized, flexible, and efficient code. Here are the types of problems they can help with:

1. **Creating Objects:**

When we need to create objects in a smart way, like making sure only one instance exists (Singleton), or creating objects without specifying their exact classes (Factory, Builder).

2. **Structuring Code:**

For arranging classes and objects in ways that are easy to manage and extend. This is useful when dealing with different types of objects that need to work together (Adapter, Composite) or when we want to add new features to objects without changing their code (Decorator).

3. **Managing Interaction:**

When objects need to communicate or coordinate, patterns help us do this effectively. For instance, when one object needs to notify others about changes (Observer), or when we want to have different ways of performing an action (Strategy).

4. **Big Picture Organization:**

Some patterns guide us in designing the overall structure of our application. They help decide how different components interact and fit together. Think of them as blueprints for creating a well-organized application (MVC, MVVM).

5. **Handling Concurrency:**

In situations where multiple tasks need to run simultaneously, patterns can help manage this smoothly. They provide solutions for avoiding conflicts and ensuring things work together correctly (Mutex, Semaphore).

Remember, using design patterns is like following a recipe when cooking. They help us solve common problems in a proven and efficient way.