

Die **strukturellen Entwurfsmuster** (auf Englisch: **structural design patterns**) sind eine Kategorie von Entwurfsmustern, die sich auf den Aufbau und die **Beziehungen zwischen Klassen und Objekten konzentrieren**. Ihr Hauptziel ist es, **flexible und effiziente Strukturen** für die Zusammenarbeit von Klassen und Objekten zu schaffen. Dabei helfen sie, komplexe Systeme durch einfachere und besser wartbare Strukturen zu organisieren.

Warum strukturelle Entwurfsmuster?

In großen und komplexen Softwareprojekten ist es oft notwendig, viele Klassen und Objekte so zu strukturieren, dass sie effizient zusammenarbeiten, leicht erweiterbar sind und sich der Code gut warten lässt. Ohne Struktur kann der Code schnell unübersichtlich werden, insbesondere wenn viele Objekte miteinander in Beziehung stehen.

Die strukturellen Entwurfsmuster bieten Lösungen, um:

- **Kopplung** zwischen Objekten und Klassen zu reduzieren.
- **Vererbung und Zusammensetzung** auf eine sinnvolle Weise zu nutzen.
- **Wiederverwendbarkeit** von Code zu fördern.
- **Flexibilität und Erweiterbarkeit** zu verbessern.

Kategorien von strukturellen Entwurfsmustern:

Die strukturellen Entwurfsmuster sind darauf ausgelegt, das **Design** und die **Beziehungen** zwischen verschiedenen Softwarekomponenten zu verbessern. Hier sind die bekanntesten **strukturellen Entwurfsmuster**:

1. **Adapter (Adapter Pattern)**
2. **Bridge (Brückenmuster)**
3. **Composite (Kompositionsmuster)**
4. **Decorator (Dekorierer-Muster)**
5. **Facade (Fassadenmuster)**
6. **Flyweight (Fliegengewicht-Muster)**
7. **Proxy (Stellvertreter-Muster)**