

## الگوهای طراحی در تولید نرم افزار چیستند و چه کاربردی دارند؟

الگوها توصیف کننده ی مسایل متواتری هستند که در شرایط مشخصی روی می دهند. الگوها شامل راه حل پیشنهادی برای حل این مسایل نیز می باشند. علاوه بر این الگوهای طراحی نرم افزاری می توانند در مستندسازی طراحی انجام شده نیز کمک کنند تا تصویری که در ذهنمان راجع به یک سیستم و یا زیرسیستم داریم در قالب مناسب بیان کنیم. افرادی که می خواهند الگوهای ما را گسترش دهند یا پیاده سازی نمایند با مطالعه الگوها دقیقاً از دیدگاه ما نسبت به سیستم و مسأله آگاه می شوند.

علاوه بر این الگوهای طراحی پاسخ گویی به انواع نیازمندی های وظیفه مندی و غیر وظیفه مندی را میسر می سازند و به این ترتیب به یک طراحی خوب کمک می کنند. الگوها علاوه بر ارایه ی راهکارهای مناسب برای دست یابی به نیازهای وظیفه مندی، توان پاسخ به نیازهای غیر وظیفه مندی مانند تغییرپذیری، آزمون پذیری و قابلیت نگهداری را نیز دارا هستند.

لازم است توجه شود که الگوهای طراحی به حل مسأله کمک می کنند و لی راحل کامل آنرا در اختیار ما نمی گذارند، از این جهت شبیه الگوهای لباس هستند که شکل کلی از طراح را در اختیار ما می گذارند ولی برای استفاده مناسب از هر الگو لازم است طراحی بانوآوری و خلاقیت خود آن را در زمینه ی مورد نیاز خود بدوزد. بحث الگوها در مهندسی اولین بار در معماری ساختمانی مطرح شد، لکن اولین کسی که از الگوها برای تولید نرم افزار کمک گرفت فردی به نام اریک گاما است که در سال ۱۹۹۱ در تز دکترای خود به بحث و بررسی الگوها پرداخت. گاما به همراه سه نفر دیگر گروه GoF را تشکیل دادند و در کتابی الگوهای خود را توصیف کردند. این کتاب با استقبال عمومی مواجه شد و از آن پس الگوهای طراحی GoF شهرت بسیاری پیدا کرده، در گستره های مختلف علم مهندسی نرم افزار به کار گرفته شدند.

کارایی و مقبولیت الگوهای طراحی (GoF Design Pattern) به حدی رسیده است که امروزه ابزارهای حرفه ای CASE بطور مستقیم از آنها پشتیبانی می کنند و امکان اعمال الگوهای طراحی GoF را فراهم می کنند. از جمله این ابزارها می توان از Rational XDE ، Rational Rose و Borland Together نام برد.

### الگوهای پایه GoF از لحاظ هدف به سه دسته تقسیم شده اند :

#### ۱. الگوهای آفرینشی (Creational)

در فرآیندهای تولید اشیاء استفاده می شوند و عبارتند از : Singleton و Prototype ، Builder ، Factory Method

#### ۲. الگوهای ساختاری (Structural)

در ترکیب کلاس‌ها و اشیاء مورد استفاده قرار می‌گیرند و عبارتند از : Composite، Bridge، Adapter، Proxy و Flyweight، Façade، Decorator

### ۳. الگوهای رفتاری (Behavioral)

چگونگی تعامل بین کلاس‌ها یا اشیاء و نحوه‌ی توزیع مسؤلیت بین آنها را مورد بحث قرار می‌دهند و عبارتند از : Interpreter, Iterator, Mediator, Memento, Observer, State, Chain of responsibility, Command, Visitor و method Strategy, Template

شخصاً در استفاده از الگوها آنها را بسیار کاربردی یافته‌ام که هم در سرعت طراحی و هم در کیفیت آن می‌تواند بسیار مفید واقع شود.

### الگوی شیء مرکب (Composite)

#### مقصود :

تشکیل ساختاری درختی به منظور نمایش سلسله مراتب کل-جزء اشیاء به Clientها اجازه میدهد با اشیاء ساده و مرکب به شکل یکسان برخورد کنند.

#### انگیزه :

در برنامه‌های گرافیکی میتوان با ترکیب اجزای پایه‌ای، اجزای مرکب ساخت نیاز به مدل کردن اشیاء مرکب مانند اشیاء ساده

#### کاربرد :

وقتی میخواهیم سلسله مراتب کل-جزء اشیاء را نمایش دهیم وقتی که میخواهیم مشترکها تفاوتی بین اشیاء ترکیبی و اشیاء بسیط احساس نکنند.

#### شرکت کنندگان :

« جزء (Component)

« برگ (Leaf)

« شیء مرکب (Composite)

« مشتری (Client)

#### همکاری‌ها :

- مشتری‌ها واسط کلاس Component را برای تعامل با اشیاء مرکب استفاده می‌کنند.
- اگر دریافت کننده یک برگ باشد، درخواست مستقیماً پاسخ داده می‌شود.
- اگر دریافت کننده مرکب باشد، درخواست به فرزندان آن فرستاده می‌شود.