

# طراحی شی گرا

۱- برنامه نویسی شی گرا چیست؟

روشی برای برنامه نویسی است که بر پایهٔ استفاده از اشیاء (Objects) طراحی شده تا کدها ساختاریافته، قابل استفادهٔ مجدد و قابل نگهداری باشند.

-----

۲- چهار اصل اصلی در طراحی شی گرا را نام ببرید.

1. کپسوله سازی (Encapsulation)

2. وراثت (Inheritance)

3. چندریختی (Polymorphism)

4. انتزاع (Abstraction)

-----

۳- تفاوت بین کلاس و شیء چیست؟

کلاس یک الگو یا قالب است، ولی شیء نمونه‌ای واقعی از آن کلاس است.  
مثلاً: کلاس = نقشه خانه، شیء = یک خانه واقعی ساخته شده.

-----

۴- کپسوله‌سازی (Encapsulation) به چه معناست و چه مزیتی دارد؟

یعنی مخفی کردن جزئیات داخلی شیء و فقط نشان دادن بخش‌های ضروری.  
مزیت: امنیت داده‌ها و جلوگیری از دسترسی مستقیم به اطلاعات حساس.

-----

۵- وراثت (Inheritance) چیست و چه کاربردی دارد؟

روشی که یک کلاس می‌تواند ویژگی‌ها و رفتارهای کلاس دیگر را به ارث ببرد.  
کاربرد: کاهش کدنویسی تکراری و افزایش قابلیت استفاده مجدد.

-----

۶- پلی‌مورفیسم (Polymorphism) را با یک مثال ساده توضیح دهید.

یعنی یک تابع یا شیء می‌تواند شکل‌های مختلفی داشته باشد.  
مثال: تابع draw() برای کلاس‌های دایره و مربع به شکل متفاوت اجرا می‌شود، ولی نام آن یکسان است.

---

## ۷- تفاوت بین Overloading و Overriding چیست؟

**Overloading:** تعریف چند تابع با نام یکسان ولی پارامترهای متفاوت در یک کلاس.

**Overriding:** تغییر عملکرد یک تابع ارث‌برده‌شده در کلاس فرزند.

---

## ۸- چرا استفاده از Interface در طراحی شی‌گرا مفید است؟

چون به برنامه اجازه می‌دهد تا وابستگی به "رفتار" داشته باشد نه پیاده‌سازی خاص. این کار انعطاف‌پذیری و قابلیت توسعه برنامه را افزایش می‌دهد.

-----

## ۹- Composition و Aggregation چه تفاوتی با هم دارند؟

**Aggregation:** رابطهٔ "دارد" است، ولی اشیاء مستقل باقی می‌مانند.

**Composition:** رابطهٔ قوی‌تر، شیء وابسته است و بدون شیء اصلی معنی ندارد.

-----

—

۱۰- اصل Open/Closed چیست و چه کاربردی در طراحی دارد؟

این اصل می‌گوید: کلاس‌ها باید برای توسعه باز و برای تغییر بسته باشند. یعنی بتوان رفتار آن‌ها را گسترش داد بدون اینکه کد اصلی را تغییر دهیم.

---