**设计模式总览**

一. 单选题（共25题，100分）

1、（单选题）采用面向对象方法设计的应用程序的特点是：

A. 编写代码更少

B. 运行速度更快

C. 占用空间更小

**D. 可复用性更强**

【答案：D】A、B、C都不一定，D正确。面向对象设计强调代码的模块化和复用性，通过封装、继承和多态性实现代码复用和扩展。

2. （单选题）面向对象方法中类的设计**不包括**：

A. 类的组织与表示

B. 行为的组织与表示

**C. 代码的组织与表示**

D. 属性的组织与表示

【答案：C】类的设计包括：①类的组织与表示②行为的组织与表示③属性的组织与表示（文件A-面向对象设计与复用-类设计）

3. （单选题）下列关于面向对象和设计模式的描述中，**错误**的是：

**A. 一个设计良好的面向对象应用程序应具有低内聚、高耦合的特点【错误】**

B. 面向对象方法中类设计的难点在于变化的存在，例如职责的变化、实现的变化等

C. 设计模式代表了最佳的实践，提供了软件开发过程中面临的一般问题的解决方案

D. 设计模式方便开发人员之间沟通和交流，使得设计方案更加通俗易懂

【答案：A】A错误，设计良好的面向对象应用程序应具有**高内聚、低耦合**的特点。

4. （单选题）面向对象设计应支持变化，下列相关描述中，**不正确**的是：

A. 一个不考虑系统变化的设计有可能需要重新设计，重新设计的代价往往是巨大的

**B. 建议通过修改既有代码来适应变化【错误】**

C. 通过显式地指定一个类来创建对象是导致重新设计的原因之一

D. 不能方便地对类进行修改是导致重新设计的原因之一

【答案：B】

A正确。B错误，根据开闭原则，修改既有代码容易引入新的错误，面向对象设计应通过扩展来适应变化，而非修改已有代码。C正确，显式创建对象导致强耦合，难以扩展。D正确，如果类设计不灵活，难以修改，就可能需要重新设计。

5. （单选题）下列选项中最适合用**依赖关系**描述的是：

A. 丈夫和妻子

B. 公司和部门

**C. 工厂和产品**

D. 水果和苹果

【答案：C】A是关联关系（长期，相对平等）；B是组合关系（公司消失，部门也小时）；C是依赖关系，工厂依赖产品类，工厂类通过使用产品类来创建产品；D是继承关系，苹果是水果的一种。

6. （单选题）在一个课程注册系统中，定义了类CourseSchedule和类Course，其中类CourseSchedule中定义了方法add（Course c）和方法remove（Course c），则这两个类之间的关系最可能是：

A. 实现关系

B. 泛化关系

C. 组合关系

**D. 依赖关系**

【答案：D】对于D：类CourseSchedule依赖类Course（Course是CourseSchedule方法中的参数），但没有长期持有它的实例，只是临时使用。对于A，实现关系通常用于接口和它的实现类。对于B，泛化关系是指类和它的子类之间的关系。对于C，组合关系表示一种强关系，容器对象和被包含对象是不可分离的。

7. （单选题）如果类A仅在其成员函数fun中定义并使用了类B的一个对象，类A其他部分的代码都不涉及类B，那么类A与类B的关系最可能是：

**A. 依赖**

B. 关联

C. 聚合

D. 组合

【答案：A】B是A函数fun的一个局部变量，表示类A暂时使用类B，故最有可能为依赖关系；关联关系表示一个类长期持有另一个类的引用或实例；聚合关系表示一个类包含另一个类的多个实例，但它们是独立存在的；组合关系表示一个类完全包含另一个类的实例，两者共存亡。

8. （单选题）下列代码中， 体现“knows a”关系的是：

**A. class B { A \* mpA };**

B. class B { public: void Fun（ A \* pA）; };

C. class B { public: A\* Fun（）; };

D. class B: public A { public: void Fun（）; }

【答案：A】“knows a”表示的是关联关系，在代码中表现为B 类中包含 A 类的指针，故A正确；B、C为依赖关系，是“uses a”；D为继承，是“is a”

9. （单选题）下列哪种用法体现出的类A和类B之间的耦合关系最强：

**A. 类A保护继承类B**

B. 类A中声明数据成员const B& b;

C. 类A中声明数据成员B b;

D. 类A中声明成员函数B\* func（） const;

【答案：A】继承的耦合关系最强。泛化/实现>组合>聚合>关联>依赖。

10. （单选题）下列选项中属于面向对象设计原则的是：

A. 抽象

B. 封装

**C. 里氏替换**

D. 多态性

【答案：C】

11.（单选题）Open-Close原则的含义是一个软件实体：

**A. 应当对扩展开放，对修改关闭**

B. 应当对修改开放，对扩展关闭

C. 应当对继承开放，对修改关闭

D. 以上都不对

【答案：A】开闭原则的定义：对扩展开放，对修改关闭。

1. （单选题）开闭原则是面向对象可复用设计的基石，开闭原则是是指一个软件实体应当对扩展开放，对（ ）关闭。

**A. 修改**

B. 分析

C. 设计

D. 实现

【答案：A】开闭原则的定义：对扩展开放，对修改关闭。

13. （单选题）下列关于开闭原则的描述中，**不正确**的是：

A. 开闭原则的英文名称是The Open-Closed Principle，简称OCP

B. 开闭原则要求软件实体（模块，类，方法等）应该对扩展开放，对修改关闭

C. 开闭原则强调接口是稳定的，可复用的，但接口的实现是可变的

**D. 面向接口进行软件设计，可以很容易地构建100%满足开闭原则的软件系统【错误】**

【答案：D】D不正确。尽管面向接口设计有助于满足开闭原则，但在实际中要完全做到100%符合是很困难的。

14. （单选题）下列关于单一职责原则的描述中，**不正确**的是：

A. 单一职责原则的英文名称是Single Responsibility Principle，简称SRP

B. 单一职责要求一个类应该只有一个职责

C. 单一职责原则有利于对象的稳定，能降低类的复杂性

**D. 单一职责原则提高了类之间的耦合性**

【答案：D】D不正确。单一职责原则降低了类之间的耦合性。

15. （单选题）在面向对象设计原则中，哪个原则表示高层模块不应该依赖于低层模块，都应该依赖于抽象，抽象不应该依赖于细节，细节应该依赖于抽象：

A. 单一职责原则

B. 开闭原则

**C. 依赖倒置原则**

D. 迪米特原则

【答案：C】依赖倒置原则要求高层模块不应该依赖于低层模块，都应该依赖于抽象。

16.（单选题）依赖倒置原则就是要依赖于（ ），而不要依赖于实现；或者说要针对接口编程，不要针对实现编程。

A. 分析

B. 设计

C. 修改

**D. 抽象**

【答案：D】依赖倒置原则的核心思想是依赖于抽象，而不是具体的实现。这样可以提高系统的灵活性和可扩展性。

17.（单选题）下列关于面向对象复用的描述中，**不正确**的是：

A. 针对接口编程，而不是针对实现编程

B. 面向对象系统中功能复用的两种最常用技术是类继承和对象组合

**C. 通过生成子类的复用称为黑箱复用，通过组装或组合对象的复用称为白箱复用【错误】**

D. 优先使用组合，而不是继承

【答案：C】C不正确。恰好相反，**通过生成子类的复用称为白箱复用（“B复用A的功能，并且B可以了解A的内部细节”），而通过组合对象的复用称为黑箱复用（“B复用A的功能，但B无法看到A的内部细节”）。**

18.（单选题）下列关于面向对象设计基本原则的描述中，**错误**的是：

A. 所有引用基类对象的地方必须能透明地使用其公有派生类的对象

B. 永远不要让一个类存在多个改变的理由

C. 使用多个专门的接口比使用单一的总接口要好

**D. 高层模块应该依赖于低层模块【错误】**

【答案：D】根据依赖倒置原则（Dependency Inversion Principle）要求高层模块和低层模块都应该依赖于抽象，而不是彼此依赖。

19.（单选题）以设计复用为目的，采用一种良好定义、正规的、一致的方式记录的软件设计经验的是：

A. 架构

B. UML模型

**C. 设计模式**

D. 设计数据

【答案：C】C正确。设计模式是记录和复用设计经验的良好定义、正规的、一致的方法。；对于A，架构是系统的高层次设计，而不是具体的设计复用模式；对于B，UML是统一建模语言，用于描述系统的设计，但不是专门用于设计复用；对于C，设计数据不是一种用于设计复用的方法。

20.（单选题）下列关于设计模式的描述中，**正确**的是：

A. 模式的概念起源于软件业

B. 设计模式共有23种

C. 设计模式与具体的应用领域紧密相关

**D. 使用设计模式能提高软件开发效率**

【答案：D】D正确，设计模式提供了成熟的解决方案，能够提高开发效率和软件质量；

对于A，不正确，模式的概念起源于建筑学，由Christopher Alexander引入软件工程领域；对于B，设计模式共有23种：部分正确。GoF总结了23种经典的设计模式，但设计模式不限于这23种，还有更多模式被不断发现和使用；对于C，不正确，设计模式是通用的解决方案，与具体的应用领域无关。

21.（单选题）下列关于设计模式的描述中，**不正确**的是：

A. 设计模式是一套被反复使用、多数人知晓的、经过分类编目的、代码设计经验的总结

**B. 理论上，设计模式一定是最优秀的解决方案【错误】**

C. 使用设计模式能方便开发人员之间的沟通和交流

D. 使用设计模式能复用成功的设计，避免那些导致不可重用的设计方案

【答案：B】B不正确，虽然设计模式提供了经过验证的解决方案，但它们并不一定是最优秀的，在具体情况下，可能需要根据实际需求进行调整。

22.（单选题）设计模式的**两大主题**是：

A. 系统的维护与开发

B. 对象组合与类的继承

C. 系统架构与系统开发

**D. 系统复用与系统扩展**

【答案：D】D正确，设计模式的两大主题是提高系统的复用性和扩展性。

23.（单选题）设计模式具有的**优点**是：

A. 提高系统性能

B. 减少类的数量，降低系统规模

C. 减少代码开发工作量

**D. 提升软件设计的质量**

【答案：D】D正确，设计模式通过提供经过验证的解决方案，帮助提升软件设计的质量；对于A，不正确，设计模式的主要目的是提高系统的可维护性和扩展性，而不是直接提高系统性能；对于B，不正确，设计模式可能会增加类的数量，因为它们强调模块化和分离关注点；对于C，设计模式的应用可能会增加初始开发工作量，但会提高代码的可维护性和复用性。

24.（单选题）设计模式**不能**解决下列哪个问题：

A. 针对接口编程

B. 增强软件扩展性和灵活性

C. 适应需求变化

**D. 确定软件功能都正确实现【不能解决】**

【答案：D】设计模式提供的是结构和设计方面的解决方案，不能确保软件功能的正确实现。A、B、C都能实现。

25.（单选题）设计模式的关键要素**不包括**：

A. 名称

**B. 目的【不包括】**

C. 解决方案

D. 效果

【答案：B】设计模式的关键要素包括①模式名称 （Pattern name） ②问题 （Problem） ③解决方案 （Solution） ④效果 （Consequences）

**创建型模式**

一、单选题（共15题）

1.（单选题）下列关于创建型模式的描述中，**错误**的是：

A. 创建型模式关注的是对象的创建

B. 创建型模式分离了对象创建和对象使用

**C. 创建型模式中客户程序必须知道要创建对象的所属具体类【错误】**

D. 创建型模式隐藏了对象实例是如何被创建和组织的

【答案：C】C不正确。创建型模式的一个目的是隐藏对象创建的细节，客户程序不需要知道具体的类。

2.（单选题）在简单工厂模式中，如果需要增加新的具体产品，通常需要修改（）的源代码。

A. 抽象产品类

B. 具体产品类

**C. 工厂类**

D. 客户类

【答案：C】简单工厂模式中，工厂类负责创建具体产品对象，如果需要增加新的具体产品，通常需要修改工厂类。

3.（单选题）以下关于工厂方法模式的叙述，**错误**的是（ ）。

A. 在工厂方法模式中引入了抽象工厂类，而具体产品的创建延迟到具体工厂中实现

B. 工厂方法模式添加新的产品对象很容易，无须对原有系统进行修改，符合开闭原则

C. 工厂方法模式存在的问题是在添加新产品时需要编写新的产品类，而且还要提供与之对应的具体工厂类，随着类个数的增加会给系统带来一些额外开销

**D. 工厂方法模式是所有形式的工厂模式中最为抽象和最具一般性的一种形态，工厂方法模式退化后可以演变成抽象工厂模式【错误】**

【答案：D】D不正确。工厂方法模式是具体工厂的一种实现方式，而抽象工厂模式是一种更高级的模式，它提供了一个用于创建一系列相关对象的接口。

4.（单选题）下列关于工厂方法模式的描述中，**不正确**的是：

A. 工厂方法模式允许系统在不修改已有工厂类的情况下引进新产品

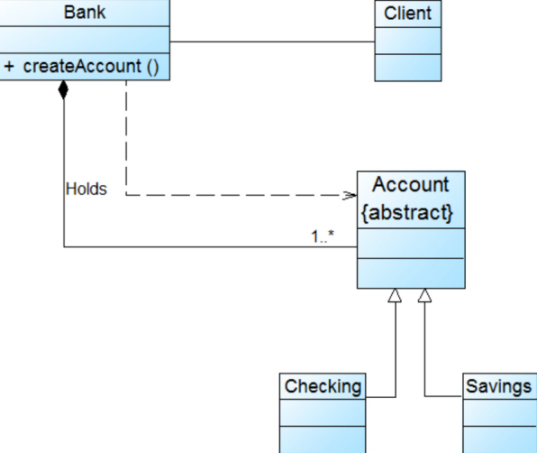
B. 工厂方法模式将产品类的实例化操作延迟到工厂子类中完成

**C. 工厂方法模式是对象创建型模式【错误】**

D. 工厂方法模式符合开闭原则

【答案：C】C错误，工厂方法模式是类创建型模式。

5.（单选题）

某银行系统采用工厂模式描述其不同账户之间的关系，设计出的类图如下所示。其中与工厂模式中的工厂角色、产品角色相对应的类分别是：

A. Client，Bank

**B. Bank，Account**

C. Bank，Checking

D. Client，Account

【答案：B】工厂依赖于产品，故选B。

6.（单选题）提供一个创建一系列相关或相互依赖对象的接口的是：

A. 工厂方法模式

**B. 抽象工厂模式**

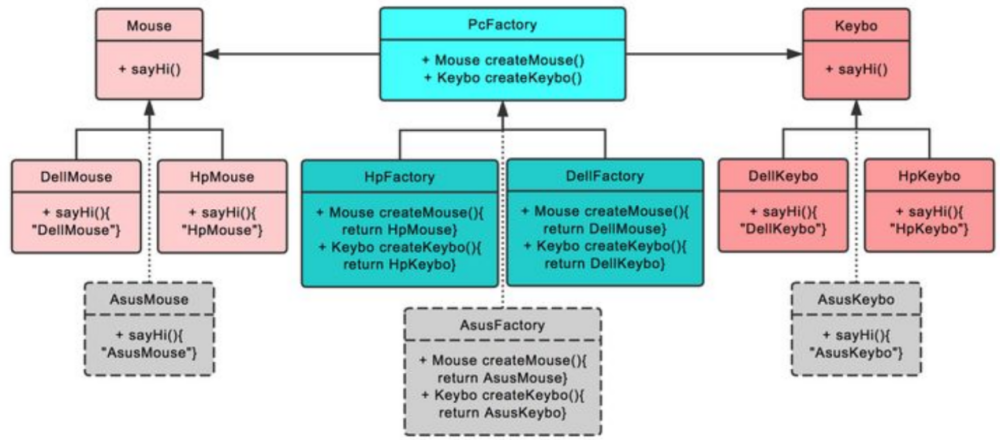
C. 生成器模式

D. 原型模式

【答案：B】看到创建，肯定是创建型模式。抽象工厂模式提供一个接口，用于创建一系列相关或相互依赖的对象（产品族），而无需指定它们的具体类，故B正确；对于A，工厂方法模式是为创建单个产品对象提供接口，而不是一系列相关对象；对于C，生成器模式用于一步一步构建复杂对象；对于D，原型模式通过复制原型对象来创建新对象。

7.（单选题）

下图描述的是那种模式?



A. 工厂方法模式

**B. 抽象工厂模式**

C. 建造者模式

D. 原型模式

【答案：B】看到Factory，大概率是某种工厂模式，同时观察到HpFactory同时依赖于HpMouse和HpKeybo两个具体产品，所以存在产品族，故为抽象工厂模式。

8.（单选题）小明正在设计一款音视频播放软件，该软件要能支持界面主题的更换，即**界面中的按钮、字体、背景等一起随界面主题的改变而变化**。针对上述需求，采用哪个设计模式最为合适：

A. 工厂方法模式

**B. 抽象工厂模式**

C. 组合模式

D. 策略模式

【答案：B】抽象工厂模式提供一个创建一系列相关或相互依赖**对象**的接口，而无需指定它们具体的类。【一个界面主题对应一个具体工厂，同时依赖于按钮、字体、背景具体的产品对象】

**为什么选择抽象工厂模式**

①一系列相关对象的创建：抽象工厂模式非常适合创建一系列相关的对象。在小明的软件中，**按钮、字体、背景等UI组件就是一系列相关的对象**，它们需要统一的创建和管理，以确保主题的一致性。【之所以不考虑状态模式，就是因为状态模式考虑的是对象在不同状态下表现出不同行为，如播放器在播放、暂停和停止状态下有不同的控制行为，例子：音乐播放器在播放状态下响应“暂停”按钮，而在暂停状态下响应“播放”按钮】

②切换主题的灵活性：抽象工厂模式允许通过实现多个具体工厂类来创建不同主题的UI组件。例如，可以有一个 LightThemeFactory 用于创建浅色主题的组件，一个 DarkThemeFactory 用于创建深色主题的组件。当需要切换主题时，只需要更换工厂即可，无需修改组件的创建逻辑。

③隐藏创建逻辑：抽象工厂模式将对象的创建逻辑封装在具体工厂中，客户端代码不需要知道具体的创建细节。这样可以降低系统的耦合性，提高代码的可维护性。

9.（单选题）某公司要开发一个图表显示系统，在该系统中曲线图生成器可以创建曲线图、曲线图图例和曲线图数据标签，柱状图生成器可以创建柱状图、柱状图图例和柱状图数据标签。用户要求可以很方便地增加新类型的图形，系统需具备良好的可扩展能力。针对这种需求，公司采用（ ）最为恰当。

A. 简单工厂模式

B. 工厂方法模式

**C. 抽象工厂模式**

D. 建造者模式

【答案：C】曲线图生成器不仅要创建曲线图，还要创建曲线图图例和曲线图数据标签。同样，柱状图生成器也需要创建柱状图、柱状图图例和柱状图数据标签。说明存在产品族和产品等级结构，故选抽象工厂模式。

10.（单选题）游戏不同场景中的房屋都由五个部分组成：地板、墙壁、窗户、门和天花板，构建房屋的步骤固定，而具体组件（门、窗等）易变。针对上述房屋，采用哪个设计模式最为合适：

A. 工厂方法

**B. 建造者**

C. 原型

D. 组合

【答案：B】建造者模式用于将一个复杂对象的构建过程与它的表示分离，使同样的构建过程可以创建不同的表示。它特别适用于需要创建具有多种复杂组件的对象，且构建步骤是固定的，但组件可以变化的情况。这里强调了“构建步骤固定”“具体组件易变”，所以为建造者模式。

11.（单选题）开发一个自动生成公文的程序，公文的基本内容包括标题、主送单位、正文、发文单位、日期及签发人等，程序应支持频繁地创建相似公文对象。采用哪个设计模式最为合适：

A. 抽象工厂

B. 建造者

**C. 原型**

D. 单例

【答案：C】原型模式（Prototype Pattern）是一种创建型模式，通过复制现有对象来创建新对象，从而避免了反复初始化对象所带来的性能开销和复杂性。它允许一个对象在运行时动态复制自身，非常适合用于需要频繁创建相似对象的场景。这里个人感觉题目中讲的应该是自动生成公文的结构，同时强调了“频繁地创建相似对象”，所以为原型模式。【类似于文件B中的“邮件复制”】

12.（单选题）某公司要开发一个即时聊天软件，用户在聊天过程中可以与多位好友同时聊天，在私聊时将产生多个聊天窗口，为了提高聊天窗口的创建效率，要求根据第一个窗口快速创建其他窗口。针对这种需求，采用（ ）进行设计最为合适。

A. 简单工厂模式

B. 抽象工厂模式

**C. 原型模式**

D. 单例模式

【答案：C】题中强调“根据第一个窗口快速创建其他窗口”，想到原型模式。尽管聊天的对象不同，但聊天窗口的布局、功能和外观通常是相似的。通过克隆现有窗口，你可以快速复制整个窗口，然后只需要更新窗口中的特定数据，如聊天对象的名称、头像和聊天记录等。

13.（单选题）限制类的实例对象只能有一个的是：

A. 工厂方法模式

B. 抽象工厂模式

C. 原型模式

**D. 单例模式**

【答案：D】单例模式是一种创建型设计模式，确保一个类只有一个实例，并提供一个全局访问点。

14.（单选题）下列关于单例模式的描述中，**不正确**的是：

A.单例类只能有一个实例

B.单例类应提供一个访问它的全局访问点

C.单例类的Instance方法是静态的

**D.单例类可以派生子类，易于扩展【错误】**

【答案：D】D不正确。单例模式的设计目的之一是确保唯一性，因此派生子类可能破坏这种唯一性，不宜扩展。（但如果将构造函数改为protected也是可以扩展的）

对于C进行扩展，**单例模式中instance为什么一定要是static的：**

①通过静态的类方法（getInstance） 获取instance，该方法是静态方法，instance由该方法返回（被该方法使用），如果instance非静态，无法被getInstance调用；

②instance需要在调用getInstance时候被初始化，只有static的成员才能在没有创建对象时进行初始化。且类的静态成员在类第一次被使用时初始化后就不会再被初始化，保证了单例。

③static类型的instance存在静态存储区，每次调用时，都指向的同一个对象。

15.（单选题）某软件系统需要从指定的XML文件读取较多的配置参数，作为全局共享资源，方便系统初始化及运行过程中使用。针对上述场景，哪个设计模式最为合适：

A.抽象工厂

B.原型

C.建造者

**D.单例**

【答案：D】单例模式确保一个类只有一个实例，并提供全局访问点。适合用于读取全局配置参数。

**结构型模式**

一. 单选题（共20题，100分）

1. （单选题）将一个类的接口转换成客户希望的另外一个接口，使得原本由于接口不兼容而不能一起工作的那些类可以一起工作的是：

A. 工厂方法模式

**B. 适配器模式**

C. 外观模式

D. 代理模式

【答案：B】适配器模式将一个类的接口转换成客户希望的另一个接口，使得原本不兼容的类能够一起工作。

2. （单选题）

下列关于适配器模式的描述中，**不正确**的是：

A. 适配器模式可以将一个接口转换成客户希望的另一个接口

B. 适配器模式中目标类（Target）可以是一个抽象类或接口，也可以是具体类

C. 适配器模式既可以作为类结构型模式，也可以作为对象结构型模式

**D. 同一个适配器可以把适配者类和它的子类都适配到目标接口【错误】**

【答案：D】D不正确。对于对象适配器，可以把适配者类和它的子类都适配到目标接口。但是对于类适配器，对于Java、C#等不支持多重继承的语言，一次最多只能适配一个适配者类，不能为具体类，其使用有一定的局限性，不能将一个适配者类和它的子类都适配到目标接口。

3. （单选题）

在对象适配器中，适配器类（Adapter）和适配者类（Adaptee）之间的关系为：

A. 依赖关系

**B. 关联关系**

C. 泛化关系

D. 实现关系

【答案：B】B正确。在对象适配器模式中，适配器类通过组合方式持有适配者类的一个实例，从而进行接口转换，所以为关联关系。【类适配器则为继承关系】

4. （单选题）现在需要开发一个文件转换软件，将文件由一种格式转换成另一种格式，例如将XML文件转换成PDF文件，将DOC文件转换成TXT文件，有些文件格式转换代码已经存在，为了将已有的代码应用于新软件而不需要修改软件的整体结构，可以使用：

A. 建造者模式

B. 原型模式

**C. 适配器模式**

D. 桥接模式

【答案：C】现有的文件格式转换代码已经存在，但它们的接口可能与新软件的需求不兼容。适配器模式可以将现有的文件格式转换代码的接口转换为新软件所需的接口，这样可以避免对现有代码进行修改。【题中强调不修改代码实现复用，想到适配器模式的“转换匹配，复用功能”】

5. （单选题）用于分离接口和具体实现，使得接口和实现可独立变化的是：

A. 适配器模式

**B. 桥接模式**

C. 组合模式

D. 外观模式

【答案：B】桥接模式用于分离抽象和实现，使得它们可以独立变化。桥接模式可以用在需要多个维度变化的场景，比如图形绘制系统中，将图形抽象和绘制实现分离。

6. （单选题）桥接模式的本质是：

A. 转换匹配，复用功能

**B. 分离抽象和实现**

C. 封装交互，简化调用

D. 控制对象访问

【答案：B】桥接模式的本质是分离抽象和实现；对于A，适配器模式的本质是转换匹配，复用功能；对于C，外观模式的本质是封装交互，简化调用；对于D，代理模式的本质是控制对象访问。

7. （单选题）

轿车可按品牌分，如红旗、奔腾、中华等，也可按变速方式来分，如手动、自动等，还可按驱动方式来分，如前驱、后驱、四驱等。针对上述轿车，采用哪个设计模式最为合适：

A. 适配器模式

**B. 桥接模式**

C. 组合模式

D. 装饰模式

【答案：B】桥接模式适用于将多个维度（如品牌、变速方式、驱动方式）分离，使它们可以独立变化和组合。

8. （单选题）

小明正在设计一个银行业务系统，该系统对于日志记录有如下要求：按格式分类需记录操作日志、 交易日志、 异常日志等； 按照距离分类分为在本地记录日志和在异地记录日志等。针对上述需求，采用哪个模式能够方便地记录各种日志：

A. 适配器模式

**B. 桥接模式**

C. 组合模式

D. 备忘录模式

【答案：B】

桥接模式适用于以下场景：

①多维度变化：当一个对象有多个变化维度，并且这些维度可以独立变化时，使用桥接模式可以将这些维度分离开来，使每个维度可以单独变化而不影响其他维度。

②避免类爆炸：通过分离抽象和实现，可以避免由于多维度变化导致类的数量急剧增加。

本题中日志记录有两个独立的维度：日志类型和记录位置。桥接模式非常适合处理这种多维度变化的情况。

9. （单选题）

可以用来描述树形结构的是：

A. 适配器模式

B. 桥接模式

**C. 组合模式**

D. 享元模式

【答案：C】正确。组合模式用于将对象组合成树形结构，表示部分-整体的层次结构。

10. （单选题）

为了使客户端以一致的方式处理树形结构中的叶子节点和容器节点，实现客户端的透明操作，组合模式中引入了：

A. 客户类

B. 叶子构件类

C. 容器构件类

**D. 抽象构件类**

【答案：D】D正确。组合模式中引入抽象构件类，使得叶子节点和容器节点实现相同的接口，从而实现客户端的透明操作。

11. （单选题）用于为一个对象添加更多功能而不使用子类的是：

A. 适配器模式

B. 桥接模式

**C. 装饰模式**

D. 外观模式

【答案：C】装饰模式通过将对象包装在装饰器类中，可以动态地为对象添加功能，而不需要使用子类。

12. （单选题）下列关于外观模式的描述中，**不正确**的是：

A. 外观模式为子系统中的一组接口提供一个统一的入口

**B. 外观模式是迪米特原则的一种具体实现，同时也完全符合开闭原则【错误】**

C. 子系统可以是一个类、一个功能模块、系统的一个组成部分或者一个完整的系统

D. 在层次化结构的系统中，可以使用外观模式定义系统中每一层的入口

【答案：B】B不正确。虽然外观模式是迪米特原则的一种具体实现，但它不一定完全符合开闭原则，因为添加新的子系统功能可能需要修改外观类。

13. （单选题）下列哪个模式是迪米特原则的典型应用：

A. 原型模式

B. 装饰模式

**C. 外观模式**

D. 代理模式

【答案：C】C正确。外观模式通过提供一个统一的接口，减少了对象之间的依赖，是迪米特原则的典型应用。

14. （单选题）小明正在维护一个遗留的大型系统，该系统设计粗糙且遗留代码高度复杂，但新需求的开发却必须依赖于它所包含的一些功能，针对上述需求，采用哪个设计模式最为合适：

A. 适配器模式

B. 装饰模式

**C. 外观模式**

D. 享元模式

【答案：C】C正确。外观模式可以为复杂子系统提供一个统一的接口，简化新需求对旧系统的调用。【本题强调“系统设计粗糙且遗留代码高度复杂”，表明子系统复杂】

15. （单选题）设计一个模块M，为系统中其他模块提供访问不同数据库的通用接口，这些数据库的访问接口有一定的差异，但访问过程相同，例如，先连接数据库，再打开数据库，最后对数据库进行查询。针对上述模块M，采用哪个设计模式最为合适：

A. 桥接模式

B. 装饰模式

**C. 外观模式**

D. 代理模式

【答案：C】C正确，外观模式可以为不同数据库的访问提供一个统一的接口，简化访问过程。客户端（系统中的其他模块）只需通过外观接口即可完成对数据库的访问操作，而不需要关心具体的实现细节。

16. （单选题）在构建一个层次结构的系统时，可以使用下列哪个模式来定义系统中每层的入口点：

A. 适配器模式

B. 桥接模式

C. 组合模式

**D. 外观模式**

【答案：D】D正确，外观模式通过提供一个统一的接口，使得系统中的每一层都可以有一个简单的入口点，方便使用和管理。

17. （单选题）小明要编写一个鼠标单击点破泡泡的解压小游戏，假设共有66个泡泡，每个泡泡的大小都一样，颜色随机，但一定是5种指定颜色中的1种，位置随机，那么设计泡泡时最适合使用：

A. 原型模式

B. 单例模式

C. 组合模式

**D. 享元模式**

【答案：D】D正确，享元模式通过共享对象来减少内存开销，特别适用于大量相似对象的场景。对于泡泡这种颜色随机但数量庞大的对象，可以通过共享颜色来节省内存。

18. （单选题）下列关于代理模式的描述中，**不正确**的是：

A. 代理模式为其他对象提供一种代理以控制对这个对象的访问

B. 代理模式包括抽象主题（Subject）、代理主题（Proxy）和真实主题（RealSubject） 3个角色

C. 代理模式有很多种类，包括远程代理、虚拟代理、保护代理和智能引用代理等

**D. 代理模式中客户端需要知道真实主题（RealSubject）对象【错误】**

【答案：D】D不正确，代理模式的一个优点是客户端不需要知道真实主题的存在，只需与代理交互即可。

19. （单选题）代理模式的本质是：

A. 转换匹配，复用功能

B. 分离抽象和实现

C. 封装交互，简化调用

**D. 控制对象访问**

【答案：D】代理模式的本质是控制对象访问

20. （单选题）计算机使用者一般会在Windows系统桌面上设置常用软件的快捷方式，以快速方便地启动软件。针对上述场景，哪个设计模式最为合适：

A. 桥接模式

B. 组合模式

C. 装饰模式

**D. 代理模式**

【答案：D】D正确，快捷方式实际上是一个代理，它为实际的应用程序提供了一个访问入口，用户通过快捷方式间接访问应用程序，这正是代理模式的典型应用。

21.桌面快捷方式是( )模式的应用实例。

**A. 代理 (Proxy)**  B. 组合 (Composite)

C. 装饰 (Decorator) D. 外观 (Facade)

【答案：A】

**行为型模式**

一. 单选题（共15题，100分）

1. （单选题）下列关于职责链模式的描述中，**不正确**的是：

A. 职责链模式可以避免请求发送者与处理者耦合在一起

B. 职责链模式可以将请求的处理者组织成一条链，并让请求沿着链传递

**C. 发出这个请求的客户端知道链上的哪一个处理者最终处理了这个请求【错误】**

D. 系统可以动态地重新组织链和分配责任

【答案：C】C不正确，职责链模式中的请求发送者不需要知道哪个处理者最终处理了请求，这样可以降低耦合性。

2. （单选题）

下列关于命令模式的描述中，**不正确**的是：

A. 命令模式又称为动作（Action）模式或事务（Transaction）模式

B. 命令模式对请求进行封装，将发出请求的职责和执行请求的职责分隔开

**C. 宏命令是命令模式和装饰模式联用的产物【错误】**

D. 命令模式可能会导致某些系统有过多的具体命令类

【答案：C】C不正确，宏命令是命令模式的一种扩展，通过将多个命令组合成一个复杂命令，而不是与装饰模式联用的产物。

3. （单选题）Web开发人员对Web服务器管理的所有Web资源：例如JSP，Servlet，静态图片文件或静态HTML文件等进行拦截，从而实现一些特殊的功能，例如实现URL级别的权限访问控制，过滤敏感词汇，压缩响应信息等，针对上述需求，采用哪个模式最为合适：

A. 原型模式

B. 装饰模式

**C. 职责链模式**

D. 中介者模式

【答案：C】在Web开发中，使用过滤器链对请求数据进行过滤和预处理，例如在Servlet中配置多个过滤器形成过滤链。职责链模式可以将处理者组织成链，使得请求沿链传递，适合用于实现URL级别的权限访问控制和其他拦截功能。

4. （单选题）体现“集中管理多个对象间的交互过程和顺序”的是：

A. 外观模式

B. 策略模式

**C. 中介者模式**

D. 状态模式

【答案：C】C正确，中介者模式通过一个中介者对象来集中管理多个对象间的交互和顺序，减少对象间的直接依赖。【注意是“多个对象间”】

5. （单选题）下列关于中介者模式的描述中，**不正确**的是：

A. 中介者通过封装交互，使得各同事对象不再需要显式地相互引用，从而使其耦合松散

**B. 中介者模式用中介者和同事的多对多交互代替了原来同事之间的一对多交互【错误】**

C. 中介者和外观模式都是迪米特法则的典型应用

D. 外观模式封装了单向交互，而中介者模式封装了多向交互

【答案：B】B不正确，中介者模式实际上是通过中介者和同事之间的一对多交互来替代原来同事之间的多对多交互，从而简化了交互关系。

6. （单选题）个人计算机由主板、CPU、内存、显卡、声卡、网卡和硬盘等配件组装而成，各个配件的交互一般通过主板来完成，针对上述场景，哪个设计模式最为合适：

A. 命令

**B. 中介者**

C. 备忘录

D. 观察者

【答案：B】B正确，中介者模式通过一个中介者（主板）管理各个配件之间的交互，减少配件之间的直接耦合。

7. （单选题）

小明正在设计一个通用数据处理软件，需要支持多个系统之间的数据传递与交换。该软件要求能够屏蔽各系统之间的直接数据交互，使其耦合松散，并且可以独立改变系统之间的交互过程。针对上述需求，采用哪个模式最为合适：

A. 简单工厂模式

B. 组合模式

**C. 中介者模式**

D. 命令模式

【答案：C】C正确，中介者模式通过一个中介者对象管理各系统之间的数据交互，减少系统之间的耦合。【注意是多个系统之间】

8. （单选题）很多游戏软件中都提供了“储存/载入进度”的功能，支持玩家在中断游戏后仍然能够重新载入被储存的进度继续游戏。针对上述场景，哪个设计模式最为合适：

A. 职责链

**B. 备忘录**

C. 状态

D. 策略

【答案：B】B正确，备忘录模式用于保存和恢复对象的状态，非常适合用于实现游戏的“储存/载入进度”功能。

9. （单选题）下列关于状态模式和策略模式的描述中**不准确**的是：

A. 策略模式和状态模式都包括环境类角色

B. 使用策略模式时，客户端需要知道所选的具体策略是哪一个

**C. 使用状态模式时，客户端需要知道环境类的状态是如何切换的**

D. 状态模式中，环境类和状态类之间可能存在双向的关联关系

【答案：C】C不正确，在状态模式中，状态的切换是内部实现细节，客户端无需了解具体的切换过程。

10. （单选题）要创建一个触发链，A对象行为将影响B对象，B对象行为将影响C对象……，可以使用：

A. 职责链模式

B. 命令模式

C. 迭代器模式

**D. 观察者模式**

【答案：D】D正确，观察者模式允许一个对象通知多个其他对象，形成触发链。例如，A对象通知B对象，B对象再通知C对象。【观察者的模式适用环境里面有提到】

11. （单选题）MVC架构在实现上结合了多种设计模式，其中最典型的模式应用是：

A. 职责链和建造者

B. 简单工厂和桥接

**C. 中介者和观察者**

D. 抽象工厂和组合

【答案：C】C正确，MVC架构中的控制器通常使用中介者模式协调视图和模型，而视图和模型之间的同步通常使用观察者模式。模型作为观察目标，视图作为观察者。

12. （单选题）兄弟俩要到操场上玩，临走时，跟妈妈说：“我俩玩去了，饭好了，招呼我们”。这段话所描述的场景最适合下列能哪种模式的应用场景？

A. 职责链模式

B. 中介者模式

C. 状态模式

**D. 观察者模式**

【答案：D】D正确，观察者模式用于一个对象通知多个其他对象，在这个场景中，兄弟俩相当于观察者，而妈妈是被观察者。

13. （单选题）小明正在设计一个报价管理模块，要求对普通客户或新客户报全价，对老客户报价统一折扣5%，对大客户报价统一折扣10%。针对上述需求，采用哪个设计模式最为合适：

A. 职责链模式

**B. 策略模式**

C. 状态模式

D. 模板方法模式

【答案：B】B正确，策略模式允许为不同客户选择不同的报价策略，例如普通客户、老客户和大客户分别对应不同的策略。

14. （单选题）三国演义中刘备去东吴招亲，赵云得授3个锦囊妙计，分别是找乔国老帮忙、求吴国太放行及孙夫人断后，以助刘备顺利回归。针对上述场景，采用哪个设计模式最为合适：

A. 职责链模式

B. 备忘录模式

**C. 策略模式**

D. 状态模式

【答案：C】C正确，策略模式允许根据不同情况选择不同的策略，这与赵云授予刘备的三个锦囊妙计的场景相符。

15. （单选题）下列选项中**不属于**对象模式的是：

A. 建造者模式

B. 代理模式

C. 迭代器模式

**D. 模板方法模式【不属于】**

【答案：D】模板方法模式是类模式。

**历年题**

1、下列面向对象设计基本原则的描述中，**错误**的是：

1. 模块应对扩展开放，而对修改关闭
2. 要针对接口编程，而不是针对实现编程
3. **优先使用继承，而不是组合【错误】**
4. 抽象不应该依赖于细节，细节应当依赖于抽象

【答案：C】C不正确，实际上，设计模式中更倡导“优先使用组合，而不是继承”。继承有时候会导致系统的耦合度增加，而组合则更加灵活，可以实现对象之间的动态关联。

1. 下列选项属于**设计模式主要优点**的是
2. 程序易于理解
3. 减少程序的最终代码量
4. **适应需求变化**
5. 简化软件系统的设计

【答案：C】

25.（单选题）设计模式的关键要素**不包括**：

A. 名称

B. 问题

C. 解决方案

**D. 实现**

【答案：D】设计模式的关键要素包括①模式名称 ②问题 ③解决方案 ④效果

3、描述对象所能接受的全部请求的集合的是：

（A）型构

**（B）接口**

（C）类型

（D）超类型

【答案：B】接口描述了对象所能接受的全部请求的集合。接口定义了类必须实现的方法，而不具体说明如何实现这些方法。通过接口，类可以实现多态性和灵活性，这是面向对象编程中的核心概念。

4、用于分离接口和具体实现，使得接口和实现可独立变化的是：

（A）适配器模式

**（B）桥接模式**

（C）命令模式

（D）模板方法模式

【答案：B】B正确，桥接模式分离抽象部分和实现部分，使它们可以独立变化。对于A，适配器模式用于将一个接口转换成另一个接口，不是用来分离接口和实现的；对于C，命令模式用于将请求封装成对象，不涉及分离接口和实现；对于D，模板方法模式定义了算法的骨架，并允许子类重新定义某些步骤，但不涉及接口和实现的分离。

5、有时你不得不改变一个难以修改的类。也许你需要源代码而又没有（对于商业类库就有这种情况），或者可能对类的任何改变会要求修改许多已存在的其他子类。提供在这些情况下对类进行修改的方法的模式有：

1. **适配器模式**
2. 桥接模式
3. 命令模式
4. **装饰模式**

【答案：A、D】黑皮书1.6.7

6、在创建对象时指定类名将使你受特定实现的约束而不是特定接口的约束。这会使未来的变化更复杂。要避免这种情况，应该间接地创建对象，对应的模式有：

**（A）抽象工厂**

**（B）工厂方法**

（C）命令模式

**（D）原型模式**

【答案：A、B、D】黑皮书1.6.7

7、设计模式帮你确定并不明显的抽象和描述这些抽象的对象。将实体的每一个状态描述为一个对象的模式是：

（A）抽象工厂（B）工厂方法 **（C）状态模式** （D）原型模式

【答案：C】黑皮书1.6.1

8、描述对象所能接受的全部请求的集合的是：

（A）型构 **（B）接口**（C）类型（D）超类型

【答案：B】黑皮书1.6.3

9、在面向对象系统中，对象接口与其功能实现是分离的，不同对象可以对请求做不同的实现，当给对象发送请求时，所引起的具体操作既与请求本身有关又与接受对象有关。发送给对象的请求和它的相应操作在运行时刻的连接就称之为：

（A）多态 （B）继承 **（C）动态绑定** （D）组合

【答案：C】黑皮书1.6.3

10. 关于类继承和对象组合，以下说法**正确**的是：

**（A）类继承是在编译时刻静态定义的，且可直接使用。**

**（B）无法在运行时刻改变从父类继承的实现。**

**（C）对象组合是通过获得对其他对象的引用而在运行时刻动态定义的。**

**（D）优先使用对象组合，而不是类继承。**

【答案：A、B、C、D】黑皮书1.6.5

11. 关于原型方法模式的说法，**错误**的是：

（A）便于在运行时刻更换原型对象；

（B）各产品必须实现复制的方法，如Clone方法；

（C）实现产品复制功能的难易程度，是应用该模式必须考虑的；

（D）设计时，不能同时使用生成器模式和原型方法模式；

【答案：A】

12. 适配器设计模式可以用于：

（A）将已有类的接口转换成和目标接口兼容

（B）改进系统性能

（C）将客户端代码数据转换成目标接口期望的合适的格式

（D）使所有接口不兼容类可以一起工作

【答案：C】

13、欲使类A的所有使用者都使用 A的同一个实例，应；

（A）将A标识为final

（B）将A标识为abstract

**（C）将单例模式应用于A**

（D）将备忘录模式应用于A

【答案：C】对于A，错误。final 关键字在 Java 中表示类不能被继承，但这与确保所有使用者都使用同一个实例无关；对于B，错误。abstract 关键字表示类不能被实例化，只能通过其子类来实例化，这与确保所有使用者都使用同一个实例无关。对于D，错误。备忘录模式用于捕获对象的内部状态，以便在以后可以恢复该状态。这与确保所有使用者都使用同一个实例无关。

14、类设计中，“变化是绝对的，稳定是相对的”，下列哪个**不属于**这句话中“变化”的范畴？

（A）改变函数参数的类型

（B）增加新的数据成员

**（C）改变编程语言 【不属于】**

（D）改变对象交互的过程和顺序

【答案：C】编程语言属于代码设计，不属于类设计

15、Open-Close原则的含义是一个软件实体：

**（A）应当对扩展开放，对修改关闭**

（B）应当对修改开放，对扩展关闭

（C）应当对继承开放，对修改关闭

（D）以上都不对

【答案：A】

16、下列关于继承表述**错误**的是：

（A）继承是一种通过扩展一个已有类的实现，从而获得新功能的复用方法

（B）泛化类（超类）可以显式地捕获那些公共的属性和方法。特殊类（子类）则通过附加

属性和方法来进行实现的扩展

（C）破坏了封装性，因为这会将父类的实现细节暴露给子类

**（D）继承本质上是“白盒复用”，对父类的修改，不会影响到子类【错误】**

【答案：D】前半句正确，继承本质上是“白盒复用”，因为子类知道父类的内部实现。但对父类的修改通常会影响到子类，因为子类依赖于父类的实现。【黑皮书1.6.5】

17、下列关于对象组合/聚合，说法**错误**的是：

（A）容器类能通过被包含对象的接口来对其进行访问

（B）属于黑盒复用，封装性好，因为被包含对象的内部细节对外是不可见

（C）可以在运行时将被包含对象改变成同类型对象，从而改变容器类的行为效果，但没有

改变容器类的接口

**（D）比继承关系更加灵活，代价是相比继承关系，增强了类间的耦合度【错误】**

【答案：D】对象组合/聚合比继承更加灵活且降低了类间的耦合度。继承在某种程度上会增加类之间的耦合度，因为子类依赖于父类的实现，而组合/聚合则更注重对象间的协作。对于A，**容器类**：一个类，它包含其他对象作为其成员变量或属性，负责管理这些对象的生命周期、访问和操作；被包含对象：被容器类所包含的对象，这些对象的细节通过其公开接口暴露给容器类，而实现细节被隐藏。

18、用于分离接口和具体实现，使得接口和实现可独立变化的是：

（A）适配器模式 **（B）桥接模式** （C）命令模式 （D）模板方法模式

【答案：B】

19、体现“集中管理多个对象间的交互过程和顺序”的是：

（A）生成器模式 （B）门面模式 （C）策略模式 **（D）中介者模式**

【答案：D】

20、限制类的实例对象只能有一个的是：

（A）原型方法模式 （B）工厂方法模式**（C）单件模式** （D）生成器模式

【答案：C】

21、用于为一个对象添加更多功能而不使用子类的是：

（A）桥接模式（B）适配器模式 （C）合成（组合）模式 **（D）装饰模式**

【答案：D】

3. 下列关于继承表述**错误**的是：

（A）继承是一种通过扩展一个已有类的实现，从而获得新功能的复用方法

（B）泛化类（超类）可以显式地捕获那些公共的属性和方法。特殊类（子类）则通过附加

属性和方法来进行实现的扩展

（C）破坏了封装性，因为这会将父类的实现细节暴露给子类

**（D）继承本质上是“白盒复用”，对父类的修改，不会影响到子类【错误】**

【答案：D】前半句正确，继承本质上是“白盒复用”，因为子类知道父类的内部实现。但对父类的修改通常会影响到子类，因为子类依赖于父类的实现。【黑皮书1.6.5】

4. 在面向对象设计时，应优先使用哪种方式实现既有代码的复用：

**（A）组合或聚合** （B）公有继承 （C）保护或私有继承 （D）拷贝、粘贴再修改

【答案：A】面向对象设计原则2：优先使用对象组合，而不是类继承。

5. 下列关于设计模式的说法，**错误**的是：

（A）设计模式都是开闭基本原则的宏观运用，本质上是没有任何模式的

（B）设计时可以应用已有的设计模式，也可以应用自己发现的新模式

（C）应用设计模式时，要考虑应用的具体特点，因为不存在万能的设计模式

**（D）应用设计模式，既可以增强系统的扩展性和灵活性，又可以减少最终的代码量【错误】**

【答案：D】D错误，虽然设计模式确实可以增强系统的扩展性和灵活性，但它们往往会增加代码的复杂性和数量。设计模式的主要目的是提高代码的可维护性和可扩展性，而不是减少代码量。

6. 在继承关系中，方法应向（ ）集中，数据应向（ ）集中。

**（A）父类，子类** （B）子类，父类（C）子类，子类（D）父类，父类

【答案：A】父类的方法可以被子类继承和使用，子类可以通过重载或重写这些方法来扩展或修改其行为。将数据成员集中在子类中，有助于保持父类的抽象和通用性，同时允许子类根据具体需求进行扩展。

7. 下列关于对象组合/聚合，说法**错误**的是：

（A）容器类能通过被包含对象的接口来对其进行访问

（B）属于黑盒复用，封装性好，因为被包含对象的内部细节对外是不可见

（C）可以在运行时将被包含对象改变成同类型对象，从而改变容器类的行为效果，但没有

改变容器类的接口

**（D）比继承关系更加灵活，代价是相比继承关系，增强了类间的耦合度【错误】**

【答案：D】对象组合/聚合比继承更加灵活且降低了类间的耦合度。继承在某种程度上会增加类之间的耦合度，因为子类依赖于父类的实现，而组合/聚合则更注重对象间的协作。

8. 下面关于工厂模式的说法**错误**的是：

（A）工厂模式可用于分离产品的使用与创建过程

（B）产品必须有一个唯一的公共基类

（C）工厂可以是抽象类，也可以是具体类

**（D）简单工厂模式是退化的工厂模式，它要求产品种类稳定【错误】**

【答案：D】D错误，简单工厂模式是一种简化的工厂模式，但它并不要求产品种类必须稳定。简单工厂模式可以处理多种产品类型，虽然所有产品的创建逻辑都集中在一个工厂类中，这可能在产品种类频繁变化时导致维护问题，但这并不是它的定义特性。简单工厂模式的关键在于它将对象创建逻辑集中在一个地方，而不是产品种类的稳定性。

9. 孙小美每天都要画好妆再出门，洗脸、打粉底、描眉、上眼影、画眼线、刷睫毛、打腮红、上唇彩，虽然每天都是这个流程，但总有所变化，如昨天“樱桃小口”，今天“性感美女”，明天“卧蚕眉、丹凤眼”，后天“芭比娃娃”了。下面哪种说法是**不恰当**的？

**（A）将“樱桃小口”、“芭比娃娃”等作为具体策略，从而使用策略模式比较合适【不恰当】**

（B）化妆的基本步骤是稳定的，但每个子步骤是易变的

（C）“芭比娃娃”表明变化会同时存在于多个子步骤中

（D）直接使用模版方法模式可能不合适，因为子步骤的多种组合会造成子类数量激增

【答案：A】策略模式主要用于封装一系列可以互相替换的算法或行为。虽然不同的化妆风格如“樱桃小口”或“芭比娃娃”可以看作不同的策略，但在这个场景中，**化妆的每个子步骤是变化的，不只是某个整体策略的变化**。所以策略模式并不完全合适。

10.“兄弟俩要到操场上玩，临走时，跟妈妈说：‘我俩玩去了，饭好了，招呼我们’”。这段话所描述的场景最适合下列能哪种模式的应用场景？

（A）访问者模式 （B）中介者模式 （C）策略模式 **（D）观察者模式**

【答案：D】观察者模式：描述的是一种一对多的依赖关系，一个对象状态发生改变时，所有依赖它的对象都会得到通知并自动更新。这里的场景中，兄弟俩离开前告诉妈妈“饭好了，招呼我们”就是一种观察者模式的应用，兄弟俩是观察者，妈妈是被观察者，当饭好了，妈妈就通知兄弟俩。