## 第九章适配器模式

1. **(B)将一个类的接口装换成客户希望的另一个接口，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。**
   1. 命令模式
   2. 适配器模式
   3. 策略模式
   4. 单例模式
2. **以下关于适配器模式叙述错误的是（C）**
   1. 适配器模式将一个接口转换成客户希望的另外一个接口，使得原本接口不兼容的那些类可以要一起工作。
   2. 在类适配器中Adapter和Adaptee是继承关系，而在对象适配器中Adapter和Adaptee是关联关系。
   3. 类适配器比对象适配器灵活，在java语言中可以通过类适配器一次适配多个适配者类。
   4. 适配器可以在不修改原来的适配者接口Adaptee的情况下将一个类的接口和另一个类的接口匹配起来。
3. **现需要开发一个文件转换软件，将文件由一种格式转换为另一种格式。例如将XML文件转换为PDF文件，将DOC文件转换为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修改软件的整体结构，可以使用（A）设计模式进行系统设计。**
   1. 适配器模式
   2. 组合模式
   3. 外观模式
   4. 桥接模式
4. **在对象适配器中，适配器类和适配者类之间的关系为（A）**
   1. **关联关系**
   2. 依赖关系
   3. 继承关系
   4. 实现关系
5. **在对象适配器中一个适配器能否适配多个适配者？如果能，应该如何实现？如果不能，请说明原因。如果是类适配器呢？**

在对象适配器中，适配器与适配者之间是关联关系，一个适配器能够对应多个适配者类，只需要在该适配器类中定义对多个适配者对象的引用即可；在类适配器中，适配器与适配者是继承关系，一个适配器能否适配多个适配者类取决于该编程语言是否支持多重类继承，例如 C++语言支持多重类继承则可以适配多个适配者，而 Java、C#等语言不支持多重类继承则不能适配多个适配者。