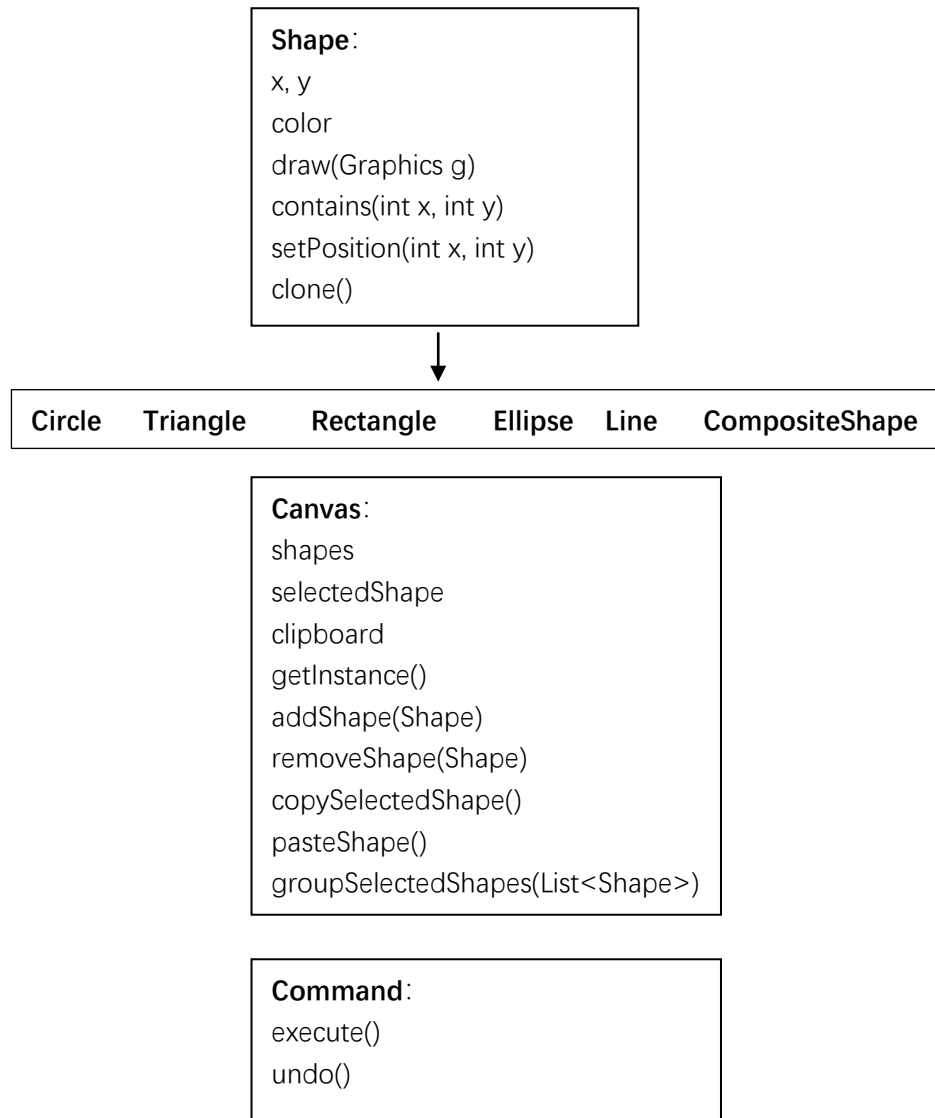


图形绘制软件使用设计模式实现

设计思想：



应用的设计模式

举例：

1、单例模式

应用位置： `Canvas.java`

单例模式确保一个类只有一个实例，并提供全局访问点。在本项目中，`Canvas` 类保证整个应用程序只有一个画布实例，避免多个画布实例带来的混乱。

2、工厂模式

应用位置： `ShapeFactory` 类

工厂模式定义一个用于创建对象的接口，但由于子类决定要实例化的类是哪一个。在本项目中，ShapeFactory 类通过接收参数来决定创建哪种图形对象，简化了对象的创建过程。

3、命令模式

应用位置：Command 类

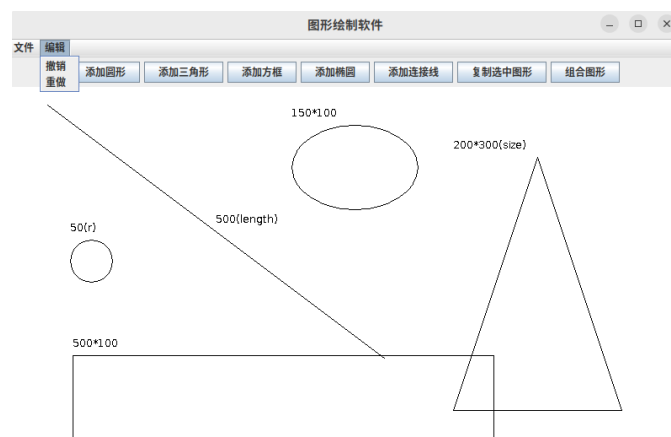
解释：命令模式将请求封装成对象，以便使用不同的请求、队列或者日志来参数化其他对象。在本项目中，命令模式用于处理图形的添加、删除等操作，支持撤销和重做功能。

4、原型模式

应用位置：Shape 类

解释：通过复制现有实例来创建新的实例。在本项目中，Shape 类及其子类都实现了 clone 方法，使得我们能够通过复制现有的图形对象来创建新对象。

演示：



sample.png



sample.gif