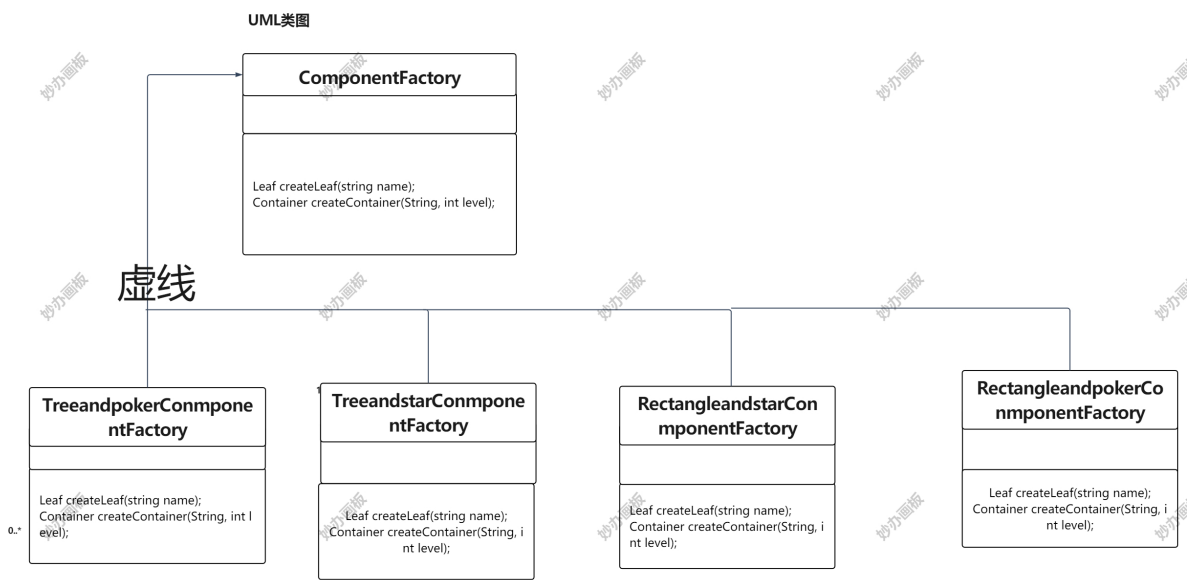


工厂方法模式：

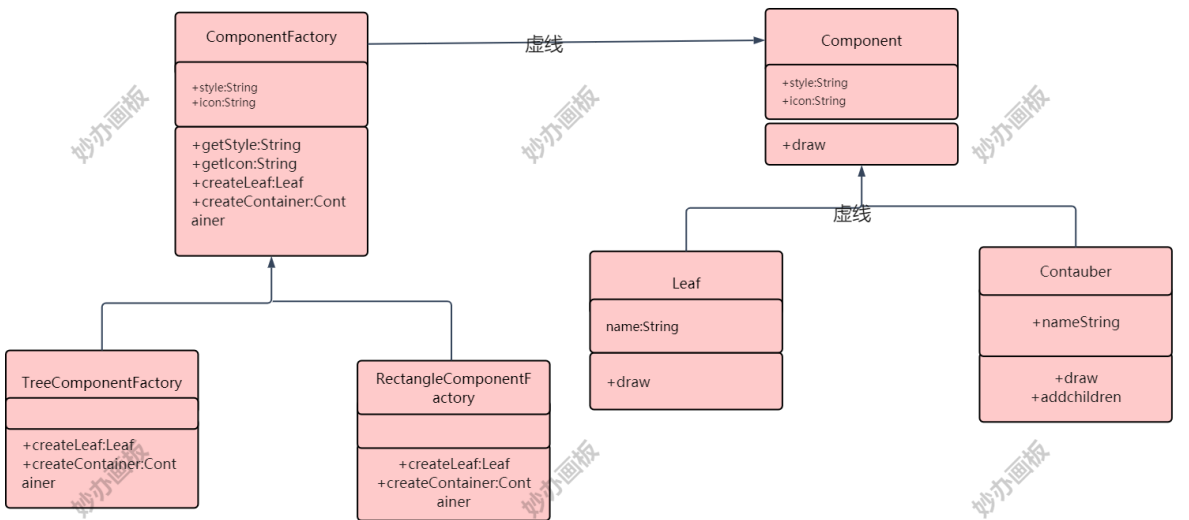
类图：



为每个style和icon新建一个工厂方法，具体的leaf和container就会根据工厂方法的设置产生。

抽象工厂模式：

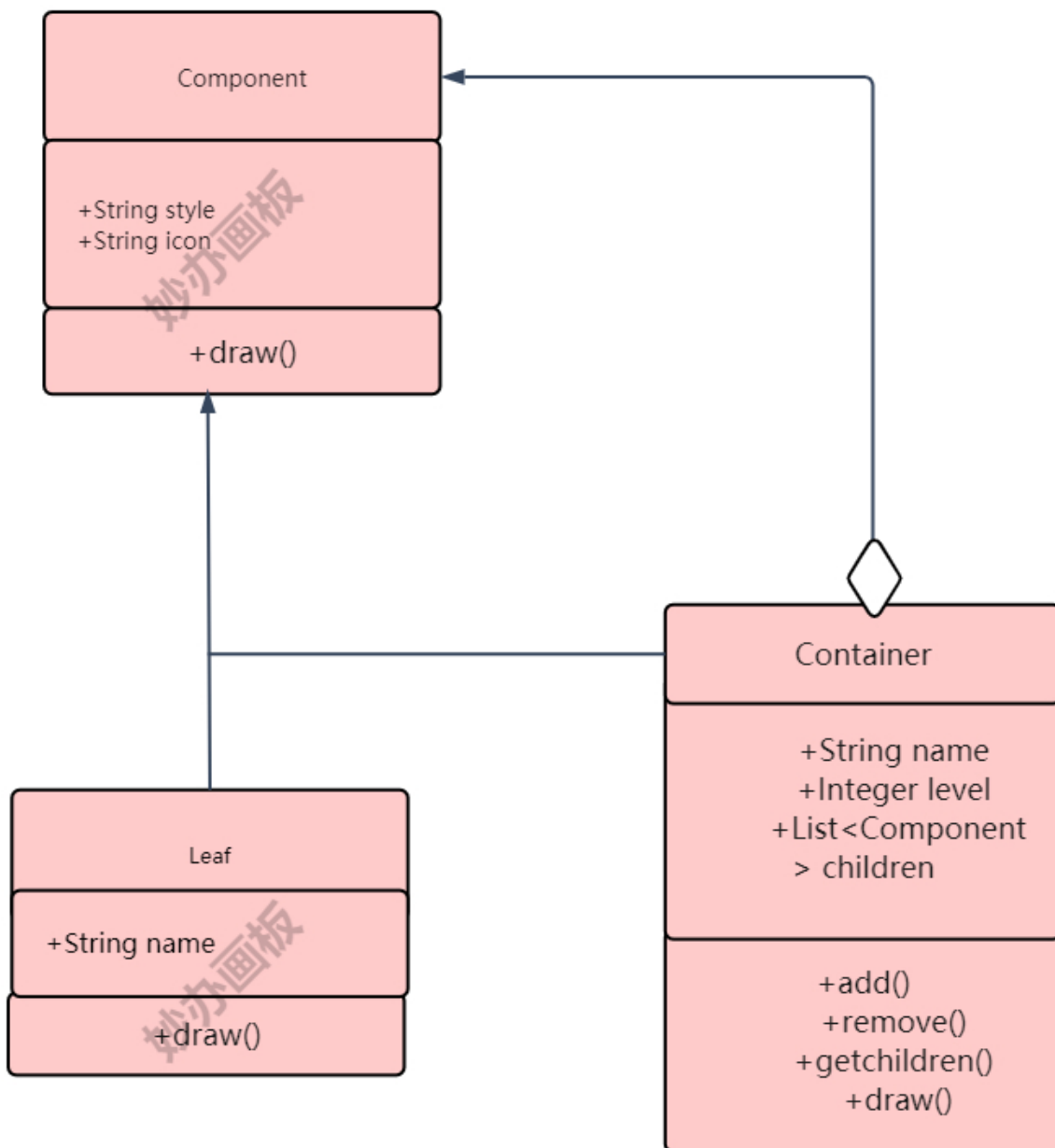
类图：



ComponentFactory作为抽象工厂，Tree和Rectangle作为两个具体工厂，Component作为抽象产品，Leaf和Contauber作为具体产品。

组合模式

类图：



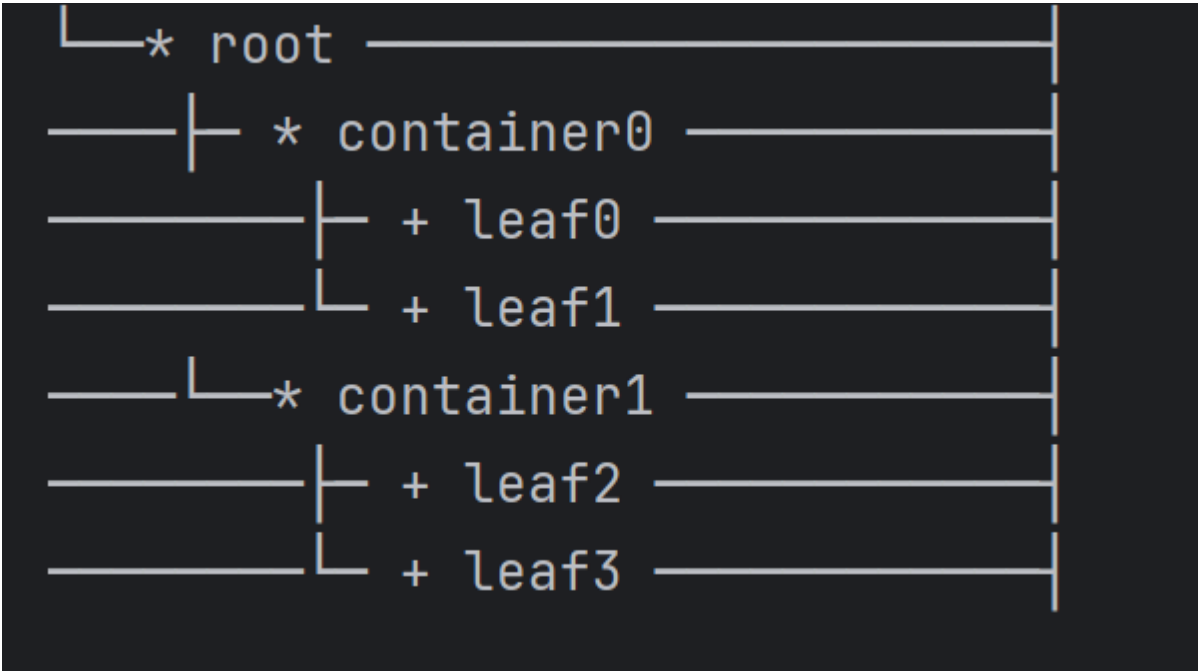
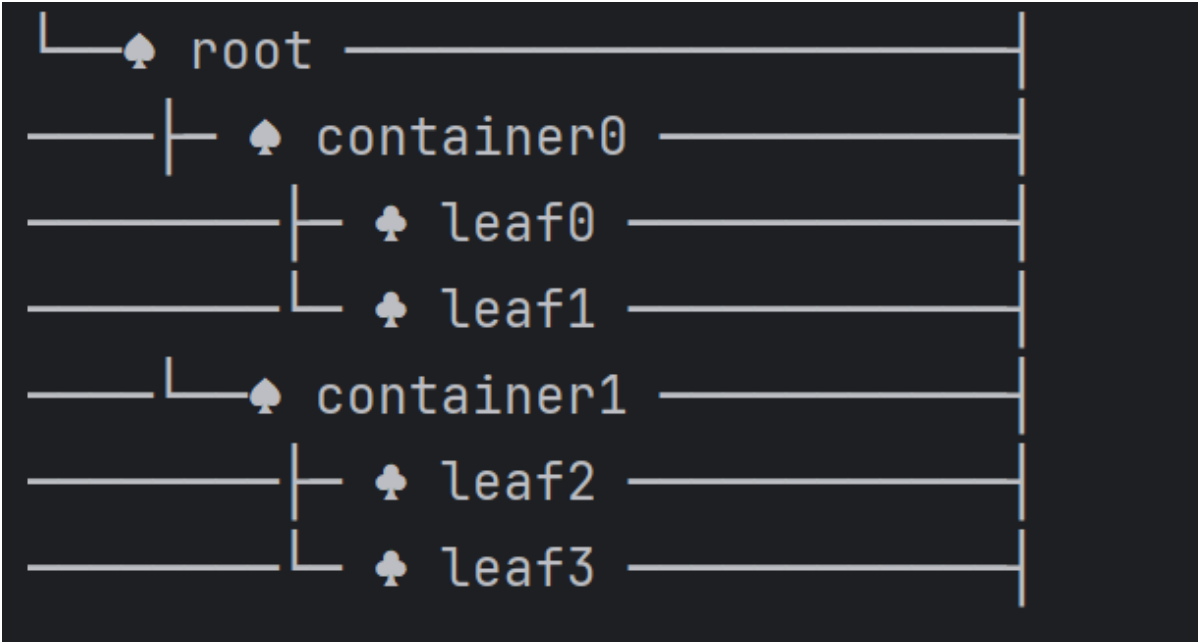
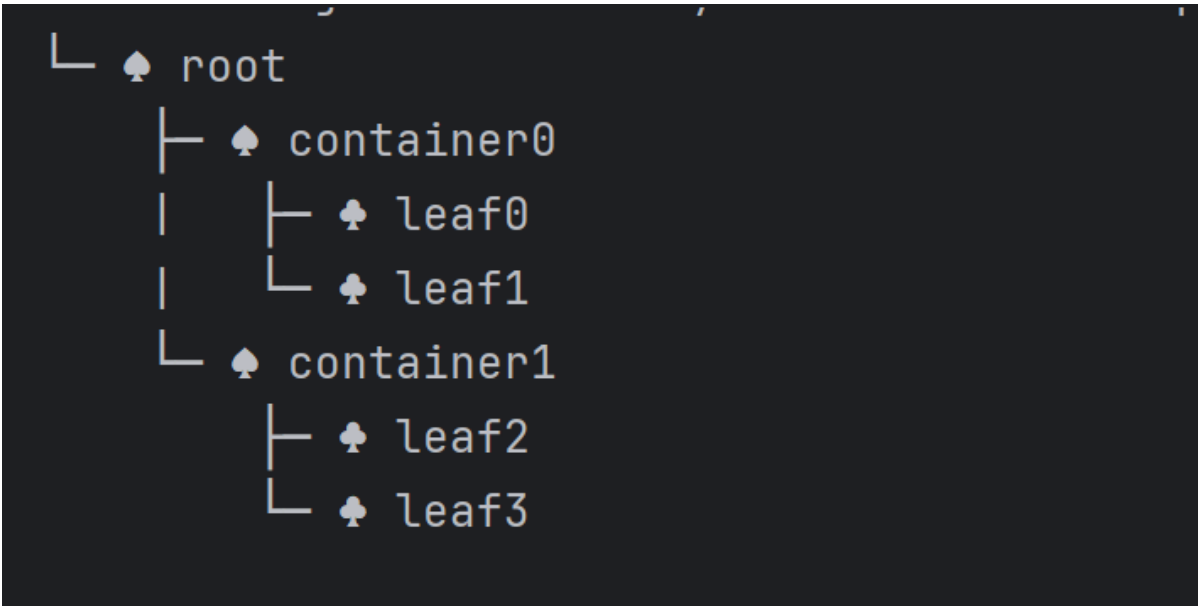
组件 (Component) 接口描述了树中简单项目和复杂项目所共有的操作。

叶节点 (Leaf) 是树的基本结构，它不包含子项目。

一般情况下，叶节点最终会完成大部分的实际工作，因为它们无法将工作指派给其他部分。

容器 (Container) ——又名“组合 (Composite)”——是包含叶节点或其他容器等子项目的单位。容器不知道其子项目所属的具体类，它只通过通用的组件接口与其子项目交互。

最终效果:



```
└ * root
  └ * container0
    |   └ + leaf0
    |   └ + leaf1
  └ * container1
    |   └ + leaf2
    |   └ + leaf3
```