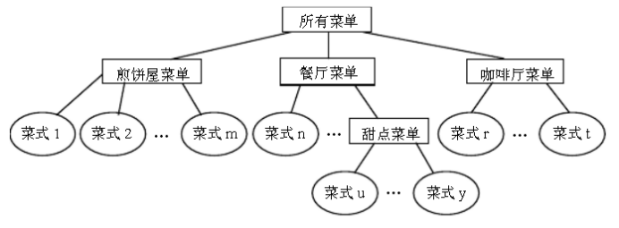
**题目：**

小王为某五星级酒店开发点餐系统。该酒店为满足客户需要，会在不同的时段提供多种不同的餐饮，其菜单的结构图如图所示。



请你采用面向对象方法通过恰当的设计模式帮助小王对上述菜单进行设计。

**解答：**

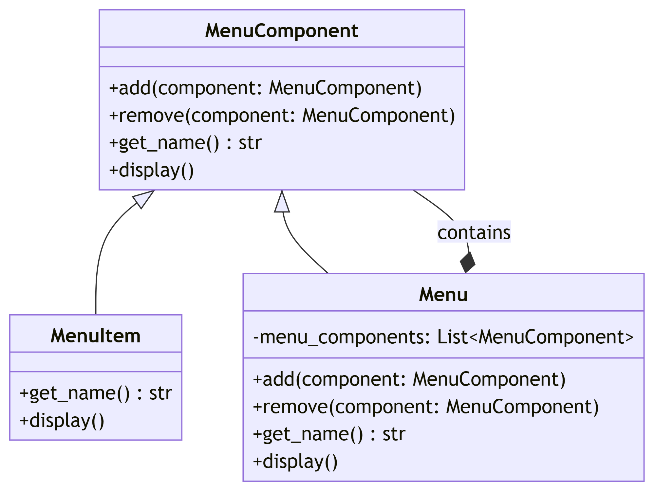
* **设计思路：**

1. 定义一个抽象的菜单组件类，用于声明所有菜单项和菜单组合的共有方法，例如添加、移除子菜单等。
2. 用菜单项表示具体的菜品，继承自菜单组件，表示不可再分的个体。
3. 用菜单组合表示一个菜单集合，继承自菜单组件，包含子菜单或菜式，可以继续包含其他菜单组合或菜单项。

* **模式：**

组合模式

* **类图：**



|  |
| --- |
| * **核心代码：** |
| from abc import ABC, abstractmethod  *# 抽象的菜单组件*  class MenuComponent(ABC):      def add(self, component):          raise NotImplementedError      def remove(self, component):          raise NotImplementedError      def get\_name(self):          raise NotImplementedError      def display(self):          raise NotImplementedError  *#具体的菜单项*  class MenuItem(MenuComponent):      def \_\_init\_\_(self, name):          self.name = name      def get\_name(self):          return self.name      def display(self):          print(f"MenuItem: {self.name}")  *# 菜单组合*  class Menu(MenuComponent):      def \_\_init\_\_(self, name):          self.name = name          self.menu\_components = []      def add(self, component: MenuComponent):          self.menu\_components.append(component)      def remove(self, component: MenuComponent):          self.menu\_components.remove(component)      def get\_name(self):          return self.name      def display(self):          print(f"Menu: {self.name}")          for component in self.menu\_components:              component.display() |