第十四周上机

窦闻

2025-5-22

1. 定义一个简单的形状继承体系，包括一个抽象基类 Shape 和几个派生类：Triangle， Rectangle，Square，Circle，每个类都实现了计算面积的方法和设置数据的方法，以及显示类名的方法。以下是各个类的详细解释：

### 基类 Shape

* **protected 成员变量**：static const char \*className，用于存储形状的类名, 初始化为 “Shape”.
* **方法**：
  + virtual float area() = 0;：纯虚函数，用于计算面积。
  + virtual void setData(float a, float b = 0) = 0;：纯虚函数，用于设置形状的尺寸。
  + virtual void showName(){ cout << "shp. " << className; }：虚函数，用于显示形状的类名。

### 类 Triangle（三角形）

* **private 成员变量**：float w, h;，分别表示三角形的底和高。
* **方法**：
  + float area() override;：重写基类的方法，计算三角形的面积。
  + void setData(float w, float h = 0) override;：重写基类的方法，设置三角形的底和高。
  + void showName() override { cout << "tri. " << className; }：重写基类的方法，显示类名。

### 类 Rectangle（矩形）

* **private 成员变量**：float w, h;，分别表示矩形的宽和高。
* **静态变量**：static const char \*className;，存储类名 “Rectangle”。
* **方法**：
  + float area() override;：重写基类的方法，计算矩形的面积。
  + void setData(float w, float h = 0) override;：重写基类的方法，设置矩形的宽和高。
  + void showName() override { cout << "rect. " << className; }：重写基类的方法，显示类名。

### 类 Square（正方形）

* **成员变量**：float s;，表示正方形的边长。
* **方法**：
  + float area() override;：重写基类的方法，计算正方形的面积。
  + void setData(float s, float = 0) override;：重写基类的方法，设置正方形的边长。
  + void showName() override { cout << "sqr " << className; }：重写基类的方法，显示类名。

### 类 Circle（圆形）

* **成员变量**：float r;，表示圆的半径。
* **方法**：
  + float area();：计算圆的面积。
  + void setData(float r, float = 0);：设置圆的半径。

### 主函数 main

* 创建各种形状的实例，并通过基类指针 Shape \*p 来操作这些实例，展示多态的使用。通过设置数据、显示类名和计算面积，演示了如何通过基类指针调用派生类的方法。示例如下：

int main()  
{  
 Shape \*p;  
 Triangle t(3, 4);  
 Rectangle r(5, 6);  
 Square s(7);  
 Circle c(8);  
   
 p = &t;  
 p->showName();  
 cout << " area: " << p->area();  
 p->setData(6, 8);  
 cout << " -> " << p->area() << endl;  
   
 p = &r;  
 p->showName();  
 cout << " area: " << p->area();  
 p->setData(10, 8);  
 cout << " -> " << p->area() << endl;  
   
 p = &s;  
 p->showName();  
 cout << " area: " << p->area();  
 p->setData(5);  
 cout << " -> " << p->area() << endl;  
   
 p = &c;  
 p->showName();  
 cout << " area: " << p->area();  
 p->setData(1);  
 cout << " -> " << p->area() << endl;  
   
 return 0;  
}

运行结果:

tri. Shape area: 6 -> 24  
rect. Rectangle area: 30 -> 80  
sqr Shape area: 49 -> 25  
shp. Shape area: 200.96 -> 3.14

1. 上一题主函数中有大量重复代码, 请抽象为函数（传指针和传引用）并在主函数中调用。