## 作业五：多态性实现

题目:

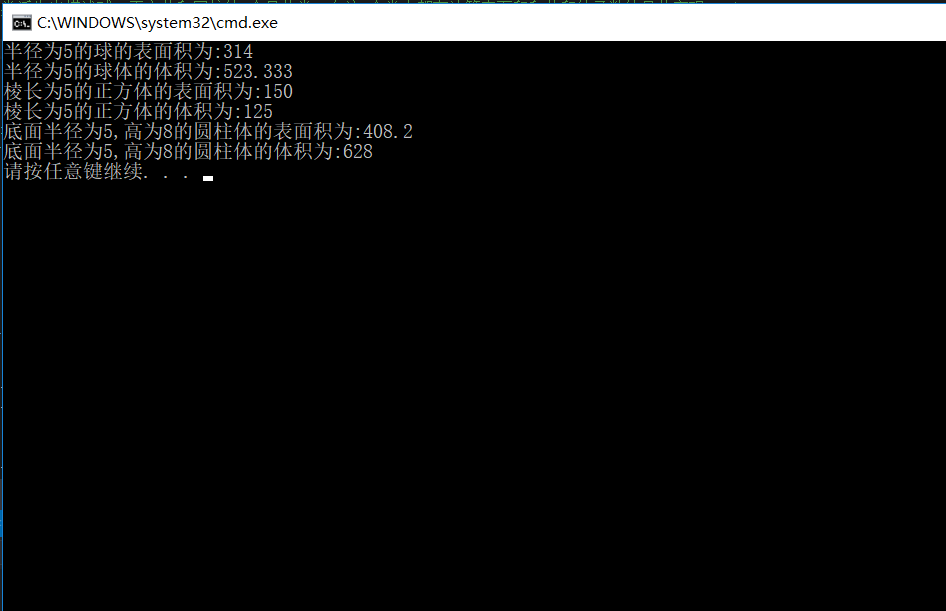
编程计算正方体、圆柱体和球的表面积和体积。

要求抽象出一个公共的基类Body，把它作为抽象类，在该类中定义求表面积和体积的纯虚函数。

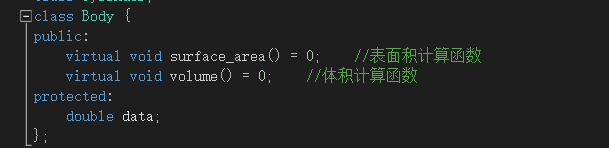
抽象类中定义一个数据成员data，它可以作为球的半径、正方体的边长或圆柱体低面圆的半径。

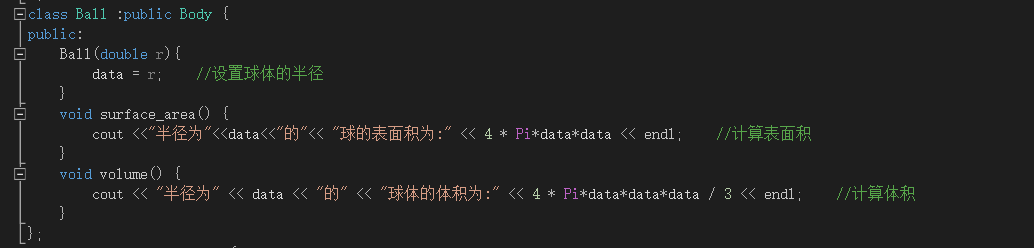
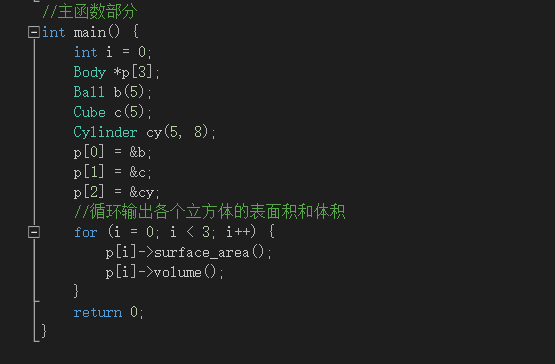
由这个抽象类派生出描述球、正方体和圆柱的3个具体类，在这3个类中都有计算表面积和体积的函数的具体实现。

结果截图：



关键部分代码：



完整代码:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*编程计算正方体、圆柱体和球的表面积和体积。

要求抽象出一个公共的基类Body，把它作为抽象类，在该类中定义求表面积和体积的纯虚函数。

抽象类中定义一个数据成员data，它可以作为球的半径、正方体的边长或圆柱体低面圆的半径。

由这个抽象类派生出描述球、正方体和圆柱的3个具体类，在这3个类中都有计算表面积和体积的函数的具体实现。\*/

#include<iostream>

using namespace std;

#define Pi 3.14 //宏定义π的值

class Ball; //对类进行声明

class Cube;

class Cylinder;

class Body {

public:

virtual void surface\_area() = 0; //表面积计算函数

virtual void volume() = 0; //体积计算函数

protected:

double data;

};

class Ball :public Body {

public:

Ball(double r){

data = r; //设置球体的半径

}

void surface\_area() {

cout <<"半径为"<<data<<"的"<< "球的表面积为:" << 4 \* Pi\*data\*data << endl; //计算表面积

}

void volume() {

cout << "半径为" << data << "的" << "球体的体积为:" << 4 \* Pi\*data\*data\*data / 3 << endl; //计算体积

}

};

class Cube : public Body {

public:

Cube(double a) {

data = a; //设置正方体的棱长

}

void surface\_area() {

cout << "棱长为" << data << "的" << "正方体的表面积为:" << 6\*data\*data << endl;

}

void volume() {

cout << "棱长为" << data << "的" << "正方体的体积为:" << data\*data\*data << endl;

}

};

class Cylinder :public Body {

public:

Cylinder(double r, double h) {

data = r; //设置圆柱体的底面半径

height = h; //设置圆柱体的高

}

void surface\_area() {

cout <<"底面半径为"<<data<<",高为"<<height<<"的"<< "圆柱体的表面积为:" << 2 \* Pi\*data\*data + 2 \* Pi\*data\*height << endl;

}

void volume() {

cout << "底面半径为" << data <<",高为" << height << "的" << "圆柱体的体积为:" << Pi\*data\*data\*height << endl;

}

private:

double height;

};

//主函数部分

int main() {

int i = 0;

Body \*p[3];

Ball b(5);

Cube c(5);

Cylinder cy(5, 8);

p[0] = &b;

p[1] = &c;

p[2] = &cy;

//循环输出各个立方体的表面积和体积

for (i = 0; i < 3; i++) {

p[i]->surface\_area();

p[i]->volume();

}

return 0;

}