

Write a program with classes to represent circle, rectangle and triangle. Each classes should have data members to represent the actual objects and member functions to read and display objects, find perimeter and area of the objects and other useful functions. Use the classes to create objects in your program.

```
// header file for input and output operation
#include <iostream>
#include <math.h>
// using standard namespace
using namespace std;
#define SUCCESS 0
#define PIE 3.14

/**
 * circle class
 *
 * \data members float radius
 * \members perimeter
 * \members area
 */
class Circle
{
private:
    float radius;
public:
    bool set(float r)
    {
        // data validation
        if(r > 0)
        {
            radius = r;
            return true;
        }
        else
        {
            cerr << "Radius must be greater than 0" << endl;
            return false;
        }
    }
    void read()
```

```

{
    int r;
    cout << "Circle" << endl;
    do
    {
        cout << "Enter the radius of the circle";
        cin >> r;
    }
    while(set(r) == false);
}
float perimeter(){return 2 * PIE * radius;}
float area(){return PIE * radius * radius ;}
void display()
{
    cout << "Circle" << endl;
    cout << "Radius :" << radius << endl;
    cout << "Perimeter :" << perimeter() << endl;
    cout << "Area :"<< area() << endl;
}
};
/**
 *
 * Class rectangle
 *
 * \data members float length
 * \data members float breadth
 * \members perimeter
 * \members area
 */
class Rectangle
{
private:
    float length;
    float breadth;
public:
    bool set(float l, float b)
    {
        // data validation
        if (l <= 0 || b <= 0)
        {

```

```

        cerr << "length and breadth should be positive" << endl;
        return false;
    }
    else
    {
        length = l;
        breadth = b;
        return true;
    }
}
void read()
{
    float l, b;
    cout << "Rectangle" << endl;
    do
    {

        cout << "Enter length and breadth?";
        cin >> l >> b;
    }
    while (set(l,b) == false);
}
float area(){return length*breadth;}
float perimeter(){return 2*(length+breadth);}
void display()
{
    cout << "Rectangle" << endl;
    cout << "length :"<< length << endl;
    cout << "breadth :"<< breadth << endl;
    cout << "Area :"<< area() << endl;
    cout << "Perimeter :"<< perimeter() << endl;
}
};
/**
 *
 * Class Triangle
 *
 * \data members float a,b,c
 * \members area
 * \members perimeter

```

```

*/
class Triangle
{
private:
    float a, b, c;
public:
    bool set(float s1, float s2, float s3)
    {
        // triangle validation
        if (s1 < (s1+s2) && s2 < (s3+s1) && s3 < (s1+s2)) // sum of two sides of the triangle is
always greater than the third
        {
            a = s1;
            b = s2;
            c = s3;
            return true;
        }
        else
        {
            cerr << "Unvalid triangle. sum of two sides of a triangle should always be
greater than third";
            return false;
        }
    };
    void read()
    {
        float s1, s2, s3;
        cout << "Triangle" << endl;
        do
        {

            cout << "Enter sides a b and c?";
            cin >> s1 >> s2 >> s3;

        }
        while (set(s1,s2,s3) == false);
    }
    float perimeter(){return a+b+c;}
    float area()
    {
        float s = perimeter()/2;
    }
}

```

```

    return sqrt(s*(s-a)*(s-b)*(s-c));
}
void display()
{
    cout << "Triangle" << endl;
    cout << "sides a " << a << " b " << b << " c " << c << endl;
    cout << "Perimeter " << perimeter() << endl;
    cout << "Area " << area() << endl;

}
};
int main()
{
    Circle circle;
    Rectangle rectangle;
    Triangle triangle;

    circle.read();
    circle.display();
    rectangle.read();
    rectangle.display();
    triangle.read();
    triangle.display();
    return SUCCESS;
}

#include<iostream>//or
#include<math.h>
#define PI 3.14156
using namespace std;
class circle
{
private:
    float radius;
public:
    void data (float x)
    {
        radius=x;
    }
    void calculations();

```

```

};
class rectangle
{
private:
    float length,breadth;
public:
    void data (float x,float y)
    {
        length=x;
        breadth=y;
    }
    void calculations();
};
class triangle
{
private:
    float l1,l2,l3;
public:
    void data (float x,float y,float z)
    {
        l1=x;
        l2=y;
        l3=z;
    }
    void calculations();
};
int main()
{
    class circle c;
    class rectangle rec;
    class triangle t;
    float r,l,b,l1,l2,l3;
    cout<<"For circle:"<<endl<<"Enter radius:\t";
    cin>>r;
    cout<<"\n\nFor rectangle:"<<endl<<"Enter length:\t";
    cin>>l;
    cout<<"\nEnter breadth:\t";
    cin>>b;
    cout<<"\n\nFor triangle:"<<endl<<"Enter length of first side:\t";
    cin>>l1;

```

```

    cout<<"\nEnter length of second side:\t";
    cin>>l2;
    cout<<"\nEnter length of third side:\t";
    cin>>l3;
    c.data(r);
    c.calculations();
    rec.data(l,b);
    rec.calculations();
    t.data(l1,l2,l3);
    t.calculations();
    return 0;
}
void circle::calculations()
{
    cout<<"\n\nFor circle:"<<endl<<"Radius:\t"<<radius<<endl;
    cout<<"Perimeter:\t"<<(2*PI*radius)<<endl;
    cout<<"Area:\t"<<(PI*radius*radius)<<endl;
}
void rectangle::calculations()
{
    cout<<"\n\nFor rectangle:"<<endl<<"Length:\t"<<length<<endl;
    cout<<"Breadth:\t"<<breadth<<endl;
    cout<<"Perimeter:\t"<<2*(length+breadth)<<endl;
    cout<<"Area:\t"<<(length*breadth)<<endl;
}
void triangle::calculations()
{
    float area,p;
    p=(l1+l2+l3)/2;
    area=sqrt(p*(p-l1)*(p-l2)*(p-l3));
    cout<<"\n\nFor triangle:"<<endl<<"Length of first side:\t"<<l1<<endl;
    cout<<"Length of second side:\t"<<l2<<endl;
    cout<<"Length of third side:\t"<<l3<<endl;
    cout<<"Perimeter:\t"<<2*p<<endl;
    cout<<"Area=:\t"<<area<<endl;
}

```