Example: Student Class

```
#include<iostream>
using namespace std;
class Student{
private:
    string name;
    float cgpa;
public:
    void setName(string n){
        name=n;
    return;
    }
    string getName(){
    return name;
    }
    void setCgpa(float cg){
        if(cg<=4 && cg>=0){cgpa=cg;}
        else {cout<<"Invalid value for CGPA"<<endl;}
    return;
    }
    float getCgpa(){
    return cgpa;
    }

    void displayInfo(){
        cout<<name<<endl;
        cout<<cgpa<<endl;
        return;
    }

};

int main(){
    Student s1,s2;
    s1.setName("student-1");
    s1.setCgpa(3.5);

    s2.setName("student-2");
    s2.setCgpa(3.6);

    s1.displayInfo();
    s2.displayInfo();

return 0;

}
```

Example: Student

```
#include<iostream>
using namespace std;
class Student{
private:
    string name="No name set";
    int id=0;
public:
    void setName(string n){name=n;}
    string getName(){return name;}
    void setId(int i){id=i;}
    int getId(){return id;}
};
int main(){
    Student s1,s2;
    s1.setName("john");
    s1.setId(2222);
    cout<<s1.getName()<<endl;
    cout<<s1.getId()<<endl;
    cout<<s2.getName()<<endl;
    cout<<s2.getId()<<endl;
return 0;
}
```

Example: Triangle

```
#include<iostream>
#include<math.h>
using namespace std;
class Triangle{
//Encapsulating The properties of Triangle
private:
    float side1=0;
    float side2=0;
    float side3=0;
public:
    //Setter methods for setting the values of properties
    void setSide1(float v){
        //logic for handling invalid values. This can also be done for
other sides.
        if(v>=0){side1=v;}
        else cout<<"invalid Value"<<endl;
    }
    void setSide2(float v){side2=v;}
    void setSide3(float v){side3=v;}
    //for setting three properties at once
    void setSide123(float v1,float v2,float v3)
    {
        side1=v1;
        side2=v2;
        side3=v3;
    }

    //getter methods for getting the values of properties
    float getSide1(){return side1;}
    float getSide2(){return side2;}
    float getSide3(){return side3;}

    //methods for getting additional information about the object
    /*
    formula for calculating perimeter
    perimeter = a+b+c
    */
    float getPrimeter(){
        return side1+side2+side3;
    }

    /*  formula for calculating Area from Perimeter
        p=perimeter/2
        area =sqrt(p(p-a)(p-b)(p-c))
    */
    float getArea(){
        float p= getPrimeter()/2;
```

```
        return sqrt(p*(p-side1)*(p-side2)*(p-side3));
    }

};
int main(){
    //creating object t1
    Triangle t1;
    //setting the values of properties using setter method
    t1.setSide123(3,4,5);
    //getting the values of properties
    cout<<t1.getSide1()<<endl;
    cout<<t1.getSide2()<<endl;
    cout<<t1.getSide3()<<endl;

    //getting perimeter of t1 triangle
    cout<<t1.getPrimeter()<<endl;
    //getting triangle Area
    cout<<t1.getArea()<<endl;

return 0;
}
```