**LAB 14: (ASSIGNMENT # 05)**

**SUBMITTED BY: NOOR-UL-AIN(BCS223020)**

**PRACTICE TASK 1:**

**CODE:**

#include <iostream>

using namespace std;

class Square; //declaration

class Circle

{

private:

double radius;

public:

Circle(double r)

{

radius = r;

}

friend double cal\_para(const Square& sq);

friend class area;

};

class Square

{

private:

double length;

public:

Square(double len)

{

length = len;

}

friend double cal\_para(const Square& sq);

friend class area;

};

double cal\_para(const Square& sq)

{

return 4 \* sq.length;

}

class area

{

public:

static double cal\_circle\_ar(const Circle& c)

{

return 3.141592653589793 \* c.radius \* c.radius;

}

static double cal\_square\_ar(const Square& s)

{

return s.length \* s.length;

}

};

int main()

{

Circle circle(7.4);

Square square(2.9);

double sqr\_para = cal\_para(square);

cout << " Perimeter of a square: " << sqr\_para << endl;

double circle\_area = area::cal\_circle\_ar(circle);

cout << " Area of a circle: " << circle\_area << endl;

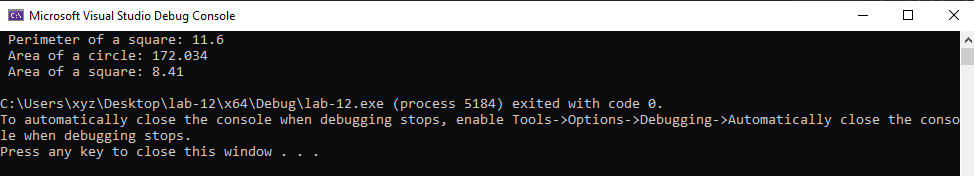
double square\_area = area::cal\_square\_ar(square);

cout << " Area of a square: " << square\_area << endl;

return 0;

}

**OUTPUT:**



**PRACTICE TASK 2:**

**CODE:**

#include <iostream>

using namespace std;

class Calculation; // declaration

class Data

{

private:

float a;

float b;

public:

Data(float x, float y)

{

a = x;

b = y;

}

friend void display(const Data& d);

friend class Calculation;

};

void display(const Data& d)

{

cout << " Data member a = " << d.a << endl;

cout << " Data member b = " << d.b << endl;

}

class Calculation

{

public:

static float sum\_sqr(const Data& data)

{

return data.a \* data.a + data.b \* data.b + 2 \* data.a \* data.b;

}

static float diff\_sqr(const Data& data) {

return data.a \* data.a + data.b \* data.b - 2 \* data.a \* data.b;

}

};

int main()

{

Data obj(2.5,5.6);

display(obj);

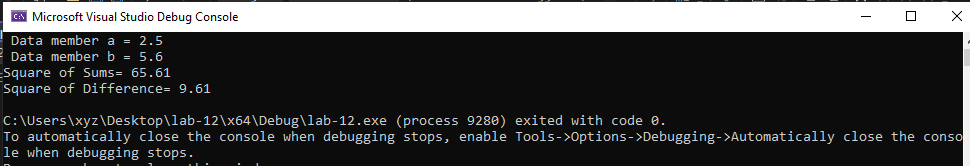
cout << "Square of Sums= " << Calculation::sum\_sqr(obj) << endl;

cout << "Square of Difference= " << Calculation::diff\_sqr(obj) << endl;

return 0;

}

**OUTPUT:**

****