National University of Computer and Emerging Sciences



Lab Exercise 12

For

Object Oriented Programming Lab

|  |  |
| --- | --- |
| Name | Muhammad Zain |
| Roll No | 19F-0228 |
| Course Instructor(s) | Dr. Danish |
| Lab Instructor(s) | Mr. Mughees Ismail |
| Semester | Spring 2020 |

|  |
| --- |
| **Question#1** |

# Source Code:

#include<iostream>

#include<string>

#include<string.h>

using namespace std;

class Polygon {

protected:

string shapeName;

public:

string toString()

{

return shapeName;

}

virtual void Area() = 0;

};

class Sphere : public Polygon

{

protected:

double Radius;

double getArea;

public:

Sphere()

{

Radius = 15;

}

void Area()

{

Polygon::shapeName = "Big Ball";

getArea = 4 \* (3.14\*(Radius\*Radius));

cout << "The Area of " << Polygon::toString() << " is " << getArea << " cm^2 " << endl;

}

};

class Rectangle : public Polygon

{

protected:

double length;

double width;

double getArea;

public:

Rectangle()

{

length = 20;

width = 35;

}

void Area()

{

Polygon::shapeName = "Deck";

getArea = length\*width;

cout << "The Area of " << Polygon::toString() << " is " << getArea << " cm^2 " << endl;

}

};

class Cylinder : public Polygon

{

private:

double Radius;

double Height;

double getArea;

public:

Cylinder() {

Height = 30;//and tank to be a cylinder of radius 10 and height 30.

Radius = 10;

} void Area()

{

Polygon::shapeName = "Tank";

getArea = 3.14\*(Radius\*Radius)\*Height;

cout << "The Area of " << Polygon::toString() << " is " << getArea << " cm^3 " << endl;

}

};

int main()

{

Rectangle Deck;

Sphere BigBall;

Cylinder Tank;

Deck.Area();

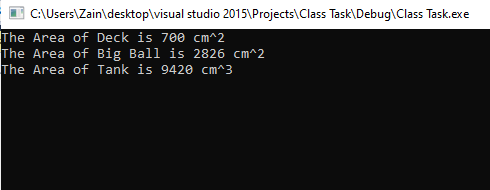
BigBall.Area();

Tank.Area();

system("pause>0");

}

# Snip:



|  |
| --- |
| **Question#2** |

# Source Code:

#include<iostream>

using namespace std;

class Rational

{

float Numerator;

float Denominator;

public:

Rational() {

cout << "This is Rational Class Constructor" << endl;

Numerator != 0;

Denominator != 0;

}

void input()

{

cout << "\tInput Rational Number "<<endl;

cout << "\n\nInput value of Numerator " << endl;

cin >> Numerator;

cout << "Input value of Denominator "<<endl;

cin >> Denominator;

while (Denominator <= 0)

{

cout << "Inavild input :-(" << endl;

cout<< "Input Again ";

cin >> Denominator;

}

}

void print1()

{

cout << "\nRationel Number before Updation " << endl;

cout << Numerator << "/" << Denominator << endl;

}

Rational operator ++()//prefix

{

Numerator++;

Denominator++;

return \*this;

}

Rational operator ++(int)

// for postfix we have to put one as argument

//ek us na khud la lena obj 2sra usko dena parna argument pass kr k

{

Numerator++;

Denominator++;

return \*this;

}

Rational operator --()//prefix

{

Numerator--;

Denominator--;

return \*this;

}

Rational operator --(int)

// for postfix we have to put one as argument

//ek us na khud la lena obj 2sra usko dena parna argument pass kr k

{

Numerator--;

Denominator--;

return \*this;

}

Rational operator !()

{

int swap;

swap = Numerator;

Numerator = Denominator;

Denominator = swap;

return \*this;

}//this to return whole function values

void Print2()

{

cout << "Rationel Number After Updation " << endl;

cout << Numerator << "/" << Denominator << endl;

}

};

int main()

{

Rational obj;

obj.input();

obj.print1();

int choice;

while(int x=1)

{

cout << "Press -1 to Exit "<<endl << endl;

cout << "Press 1 to call ++ prefix " << endl;

cout << "Press 2 to call ++ postfix " << endl;

cout << "Press 3 to call -- prefix " << endl;

cout << "Press 4 to call -- postfix " << endl;

cout << "Press 5 to call ! operator" << endl<<endl;

cin >> choice;

//system("cls");

if (choice == 1)

{

++obj;

obj.Print2();

}

else if (choice == 2)

{

obj++;

obj.Print2();

}

else if (choice == 3)

{

--obj;

obj.Print2();

}

else if (choice == 4)

{

obj--;

obj.Print2();

}

else if (choice == 5)

{

!obj;

obj.Print2();

}

else if (choice == -1)

{

break;

}

}

system("pause>0");

}

# Snip:

A screenshot of a cell phone on a table

Description automatically generatedA screenshot of a cell phone on a table

Description automatically generated

Happy Coding