ПРИКЛАД. Описти клас прямокутник у якому використати принцип інкапсуляції (властивості або методи get set).

// ConsoleApplication11.cpp : Defines the entry point for the console application.

//

#include "stdafx.h"

#include<iostream>

using namespace std;

ref class Rectangle

{

private:

double \_a, \_b;

public:

Rectangle(double a, double b){

try {

A=a;

setB(b);

}

catch (...)

{

\_a = \_b = 0;

}

}

property double A {

double get()

{

return \_a;

}

void set(double value)

{

if (value >= 0)

{

\_a = value;

}

else

throw 35;

}

}

property double Z;

double getB()

{

return \_b;

}

void setB(double value)

{

if (value >= 0)

{

\_b = value;

}

else

throw 3.5;

}

static double getArea(double a, double b) {

return a\*b;

}

double Area()

{

return \_a\*\_b;

}

property double Perimeter

{

double get()

{

return 2 \* (A+getB());

}

}

~Rectangle();

private:

};

Rectangle::~Rectangle()

{

}

int f(int a)

{

try

{

return 1 / a;

}

catch (int e)

{

cout << "div by zero" << endl;

throw e;

}

}

int main()

{

try {

cout << f(0) << endl;

}

catch (...)

{

cout << "error fro function " << endl;

}

Rectangle^ r7 = gcnew Rectangle(23, 54);

Rectangle r1(2,7);

try {

r1.A=90;

r1.setB(-23);

cout << "------------------ hello -------------------";

}

catch (int e) {

switch (e)

{

case 35:cout << "Bad value: " << e << endl;

break;

case 23:cout << "Something else" << e << endl;

break;

}

}

catch (double) {

cout << "Error double number"<<endl;

}

catch (char\* mm)

{

cout << mm << endl;

}

catch (...)

{

}

//----------------------------------

r1.setB(36);

r1.A = 89;

cout << "A=" << r1.A << endl;

Rectangle r3(2,7);cout << "S=" << r1.Area()<<endl;

cout << "S=" << Rectangle::getArea(2,7) << endl;

cout << "Perimeter=" << r1.Perimeter << endl;

system("pause");

return 0;

}