**РК-5. Папин Алексей Владимирович. ИУ5Ц-41Б.**

**Main.cpp**

// RK\_31.05.2022.cpp : Этот файл содержит функцию "main". Здесь начинается и заканчивается выполнение программы.

//

#include <iostream>

#include "Class.h"

using namespace std;

int main()

{

system("chcp 1251 >> null");

cout << "ТЕСТ: " << endl;

Flate obj(2, 4);

cout << obj << endl;

cout << "Площадь: " << obj.square() << endl;

cout << endl;

cout << "Создание объекта абстрактного класса: " << endl << endl;

Space\*\* base\_obj = new Space\*[4];

base\_obj[0] = new Room(1, 2);

Room\* first = new Room[3];

first[0] = {1, 2};

first[1] = {2, 3};

first[2] = {4, 5};

base\_obj[1] = new Flate(first, 3);

base\_obj[2] = new Room(4, 5);

Room\* second = new Room[5];

second[0] = { 6, 7 };

second[1] = { 8, 9 };

second[2] = { 10, 11 };

second[3] = { 12, 13 };

second[4] = { 14, 15 };

base\_obj[3] = new Flate(second, 5);

int sum = 0;

for (int i = 0; i < 4; i++) {

cout << endl;

base\_obj[i]->show();

cout << endl;

cout << "Площадь: " << base\_obj[i]->square() << endl;

// Сумма всех площадей

sum = sum + base\_obj[i]->square();

}

cout << "Сумма всех площадей: " << sum << endl;

// Прерывает цикл

bool check = false;

// Освобождение памяти

while (check == false) {

for (int i = 0; i < 4; i++) {

delete base\_obj[i];

cout << endl;

}

check = true;

}

}

**Class.h**

#pragma once

#include <iostream>

using namespace std;

// Абстрактный класс

class Space {

protected:

// Площадь комнаты

double S;

public:

virtual int square() = 0;

virtual void show() = 0;

~Space() { cout << "DESTRUCTION. ~Space" << endl; }

};

// Модель комнаты

class Room : public Space {

protected:

// Длина и ширина комнаты

double x, y;

// Площадь комнаты

double S;

public:

Room() { x = 0; y = 0; S = square(); }

Room(double X, double Y) { x = X; y = Y; S = square(); }

~Room() { cout << "DESTRUCTION. ~Room" << endl; }

double get\_S() { return S; }

double get\_x() { return x; }

double get\_y() { return y; }

int square() {

//cout << "Вызов виртуальной фунции. Room" << endl;

S = x \* y;

return S;

}

void show() {

cout << endl;

cout << "Комната: " << endl

<< "Длина и ширина комнаты: " << endl

<< "Длина: " << x << endl

<< "Ширина: " << y << endl;

//cout << "Площадь: " << S << endl;

}

friend ostream& operator<<(ostream& out, Room obj) {

out << "Длина и ширина комнаты: " << endl

<< "Длина: " << obj.x << endl

<< "Ширина: " << obj.y << endl

<< "Площадь: " << obj.S << endl;

return out;

}

};

// Модель квартиры

class Flate : public Room {

protected:

// Массив объектов класса Room, т.е. массив комнаты

Room\* arr;

int size;

public:

Flate() { arr = new Room; size = 0; }

Flate(Room\* ARR, int SIZE) {

size = SIZE;

arr = new Room[size];

for (int i = 0; i < size; i++) {

arr[i] = ARR[i];

}

}

Flate(double X, double Y) : Room(X, Y) {

arr = new Room[5];

size = 1;

arr[0]= Room(X, Y);

}

~Flate() { cout << "DESTRUCTION. ~Flate" << endl; }

int square() {

double sum = 0;

//cout << "Вызов виртуальной фунции. Flate" << endl;

for (int i = 0; i < size; i++) {

sum += arr[i].square();

}

return sum;

}

void show() {

for (int i = 0; i < size; i++) {

cout << endl;

cout << "Квартира: №" << i + 1<< endl;

cout << "Длина и ширина комнаты: " << endl

<< "Длина: " << arr[i].get\_x() << endl

<< "Ширина: " << arr[i].get\_y() << endl;

cout << "Площадь: " << arr[i].get\_S() << endl;

}

}

friend ostream& operator<<(ostream& out, Flate& obj) {

out << "Квартира: " << endl;

out << "Длина и ширина комнаты: " << endl

<< "Длина: " << obj.arr->get\_x() << endl

<< "Ширина: " << obj.arr->get\_y() << endl

<< "Площадь: " << obj.arr->get\_S() << endl;

return out;

}

};

**Вывод на консольное окно:**

ТЕСТ:

DESTRUCTION. ~Room

DESTRUCTION. ~Space

Квартира:

Длина и ширина комнаты:

Длина: 2

Ширина: 4

Площадь: 8

Площадь: 8

Создание объекта абстрактного класса:

DESTRUCTION. ~Room

DESTRUCTION. ~Space

DESTRUCTION. ~Room

DESTRUCTION. ~Space

DESTRUCTION. ~Room

DESTRUCTION. ~Space

DESTRUCTION. ~Room

DESTRUCTION. ~Space

DESTRUCTION. ~Room

DESTRUCTION. ~Space

DESTRUCTION. ~Room

DESTRUCTION. ~Space

DESTRUCTION. ~Room

DESTRUCTION. ~Space

DESTRUCTION. ~Room

DESTRUCTION. ~Space

Комната:

Длина и ширина комнаты:

Длина: 1

Ширина: 2

Площадь: 2

Квартира: №1

Длина и ширина комнаты:

Длина: 1

Ширина: 2

Площадь: 2

Квартира: №2

Длина и ширина комнаты:

Длина: 2

Ширина: 3

Площадь: 6

Квартира: №3

Длина и ширина комнаты:

Длина: 4

Ширина: 5

Площадь: 20

Площадь: 28

Комната:

Длина и ширина комнаты:

Длина: 4

Ширина: 5

Площадь: 20

Квартира: №1

Длина и ширина комнаты:

Длина: 6

Ширина: 7

Площадь: 42

Квартира: №2

Длина и ширина комнаты:

Длина: 8

Ширина: 9

Площадь: 72

Квартира: №3

Длина и ширина комнаты:

Длина: 10

Ширина: 11

Площадь: 110

Квартира: №4

Длина и ширина комнаты:

Длина: 12

Ширина: 13

Площадь: 156

Квартира: №5

Длина и ширина комнаты:

Длина: 14

Ширина: 15

Площадь: 210

Площадь: 590

Сумма всех площадей: 640

DESTRUCTION. ~Space

DESTRUCTION. ~Space

DESTRUCTION. ~Space

DESTRUCTION. ~Space

DESTRUCTION. ~Flate

DESTRUCTION. ~Room

DESTRUCTION. ~Space