

Лабораторна робота №7. Поліморфізм

Тема: Класи. Поліморфізм. Абстрактні класи.

Мета: Отримати знання про парадигму ООП – поліморфізм. Навчитися застосовувати отримані знання на практиці.

ВИМОГИ

1.1 Інформація про розробника:

- Кліщов Б. Р.
- КІТ 102.8а

1.2 Загальне завдання

Модернізувати попередню лабораторну роботу шляхом:

- додавання ще одного класу-спадкоємця до базового класу. Поля обрати самостійно;
- базовий клас зробити абстрактним. Додати абстрактні поля;
- розроблені класи-списки поєднуються до одного таким чином, щоб він міг працювати як з базовим класом, так і з його спадкоємцями. При цьому, серед полів класу-списку повинен бути лише один масив, що містить усі типи класів ієрархії. Оновити методи, що працюють з цим масивом.

1.3 Додаткові умови виконання завдання:

- продемонструвати відсутність витоків пам'яті;
- продемонструвати роботу розроблених методів за допомогою модульних тестів;
- не використовувати конструкцію «using namespace std;», замість цього слід робити «using» кожного необхідного класу:using std::string, using std::cout;
- в проєкті не повинні використовуватися бібліотеки введення / виведення мови C, а також не повинні використовуватися рядки типу char*.

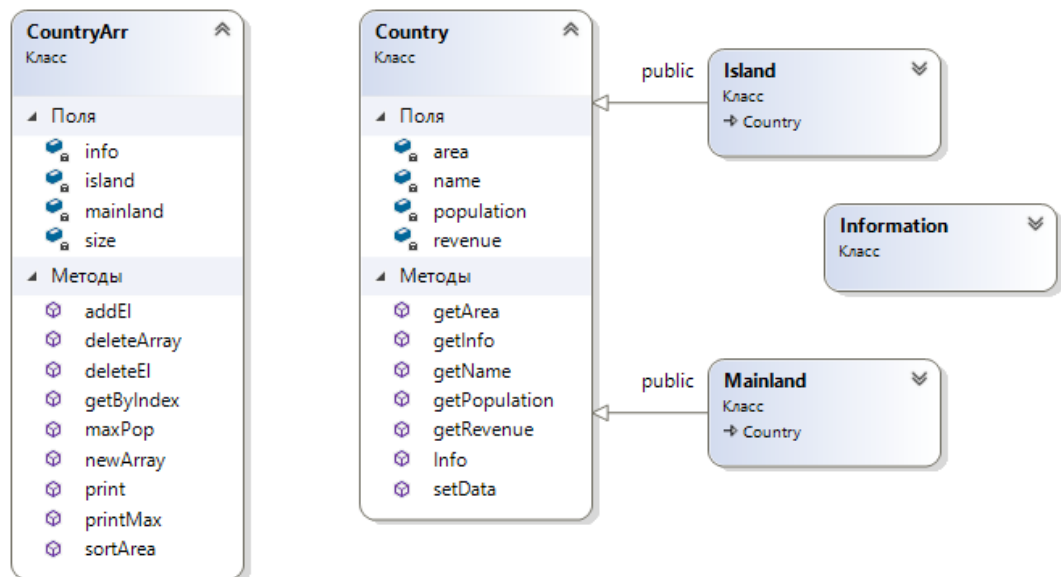
2. ОПИС ПРОГРАМИ

2.1 Функціональне призначення

Програма призначена щоб отримувати та зберігати інформацію щодо різних країн світу.

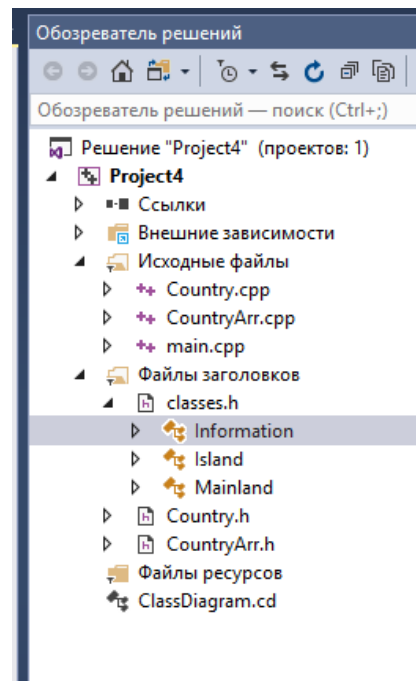
2.2 Опис логічної структури

На рисунку № 1 зображена діаграма класу



Малюнок №1. Діаграма класу

На рисунку № 2 зображена структура програми:



Малюнок №2: Структура програми

2.3 Важливі фрагменти програми

Код програми базового класу та класів спадкоємців:

```
class Island : public Country {
public:
    void Info() override {
        cout << "This country is washed by the ocean." << endl;
    }
};

class Mainland : public Country {
public:
    void Info() override {
        cout << "This country is on the mainland" << endl;
    }
};

class Information {
public:
    void Info(Country *country)
    {
        country->Info();
    }
};
```

```

class Country {
private:
    int population;
    int area;
    int revenue;
    string name;
public:
    int getPopulation();
    int getArea();
    int getRevenue();
    string getName();
    void getInfo(std::string s);
    void setData(int population, int area, int revenue, string name);
    virtual void Info() = 0;
};

```

3 ВАРІАНТИ ВИКОРИСТАННЯ

3.1 Результат роботи функцій

На рисунку № 3 зображено результат роботи програми

```

C:\Users\bogda\Desktop\нпор. лаб. 2 сем\Project07\Debu
The name of country: China
Population: 40041
Area: 48467
Revenue: 46334
This country is washed by the ocean.

The name of country: USA
Population: 3267522
Area: 2452792
Revenue: 42852
This country is washed by the ocean.

The name of country: Poland
Population: 5679762
Area: 4567899
Revenue: 65789
This country is washed by the ocean.

The name of country: Japan
Population: 735126
Area: 6832478
Revenue: 367299
This country is washed by the ocean.

Choose option:
0 - Exit

```

Рисунок № 3. Результат роботи програми

Висновок: Отримав знання про парадигму ООП – поліморфізм. Навчився застосовувати отримані знання на практиці.

