

Лабораторна робота № 2

Класи та специфікатори доступу. Інкапсуляція. Константи.

Мета: Отримати базові знання про класи, конструктори та деструктори. Дослідити механізм створення та видалення об'єктів.

1 ВИМОГИ

1.1 Розробник Інформація про розробника:

Куйдин Михаил Андреевич

НТУ “ХПІ”,

КІТ 102.8а

1.2 Завдання

Поширити попередню лабораторну роботу наступним чином:

- при введенні інформації про базовий клас (немає різниці, чи з клавіатури, чи з файлу), організувати перевірку відповідності наступним критеріям з використанням регулярних виразів:
- можна вводити тільки кириличні символи, латинські символи, цифри, пропуски, розділові знаки;
- не повинно бути пропусків та розділових знаків, які повторюються;
- перше слово не повинно починатися з маленького символу;
- в клас-списку додати метод, що виводить на екран список усіх об'єктів, які мають одне або більше полів з щонайменше двома словами (перевірку організувати за допомогою регулярних виразів).

2. Опис програми

2.1 Функціональне призначення

Програма вдосконалена за допомогою регулярних виразів.

2.2 Діаграма класів

Діаграма класів зображена на рис.1

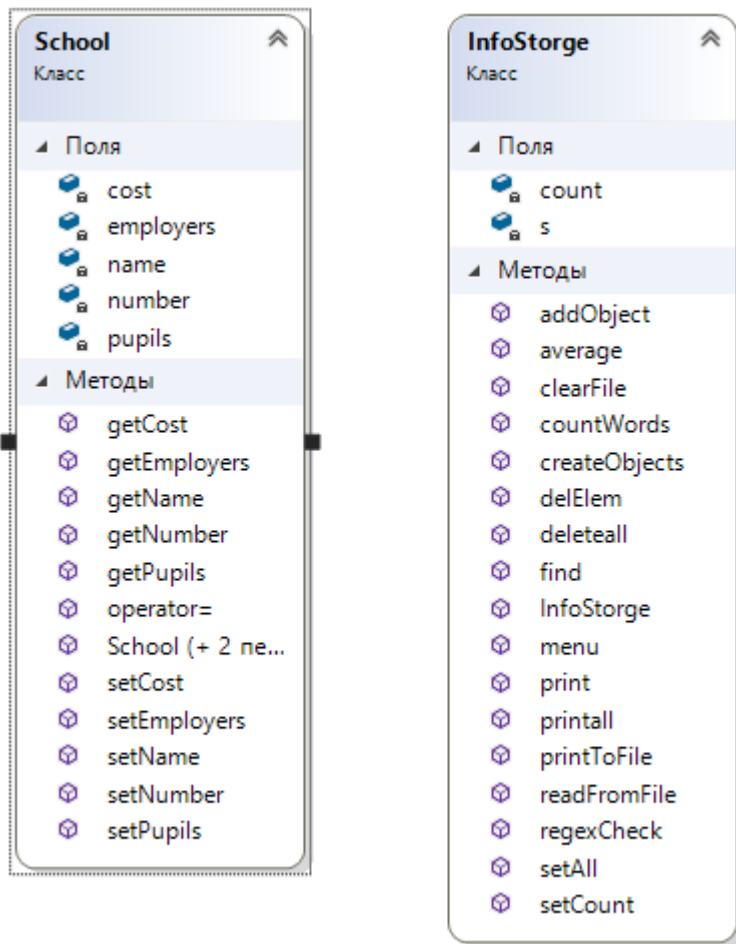


Рисунок 1

School –

InfoStorge –

2.3 Поширення програми

Додані 2 нові методи:

1. `regexCheck(string str);`

2. `countWords(int index);`

`regexCheck` – метод, який перевіряє рядок на співставлення до формату за допомогою регулярних виразів. Код зображено на рис.2

```
int InfoStorge::regexCheck(string str) {
    regex regular("^[A-Z]");
    regex space("\\s{2,}");

    while (regex_search(str, space) || !regex_search(str, regular)) {
        cout << "Name: " << str << endl;
        cout << " Name does not match the format, first letter need to be big or u put 2 of more symbols\n Enter new name that match format: ";
        return 0;
    }
    return 1;
}
```

```

num = regexCheck(p);
while (num != 1) {
    getline(cin, p);
    system("cls");
    num = regexCheck(p);
}

```

Рисунок 2

countWords – метод , який визначає кількість слів у рядку. Якщо у рядку 2 або більше слова , тоді функція повертає true, якщо менше , тоді false. Код зображено на рис.3

```

bool InfoStorge::countWords(int index) {
    regex regular("[A-Z][a-z]+\\s[a-z]+");
    string tmp = s[index].getName();
    if (regex_search(tmp, regular)) {
        return true;
    }
    return false;
}

for (int i = 0; i < tmp.count; i++) {
    result=tmp.countWords(i);
    if (result == true) {
        tmp.print(i);
    }
}

```

Рисунок 3

3 Варіанти Використання

Приклад роботи регулярних виразів зображено на рис.4.

```

In which position u want add new school? Last position:2
2
Enter cost per month: 400
Enter name of school: high
Name: high
Name does not match the format, first letter need to be big or u put 2 of more symbols
Enter new name that match format:

```

Рисунок 4

На рис.5 зображено вікно консолі , у видно що є 3 об'єкта, але тільки 2 з них мають 2 слова в назві.

```

Number of school: 179
Type of school: Kharkiv gymnasium
Pupils: 200
Cost per month: 300
Employers: 20

Number of school: 300
Type of school: High
Pupils: 400
Cost per month: 400
Employers: 20

Number of school: 179
Type of school: Kharkiv gymnasium
Pupils: 200
Cost per month: 300
Employers: 20

```

На рис.6 зображено результат роботи функції countWords та методу виводу на екран.

```
Number of school: 179
Type of school: Kharkiv gymnasium
Pupils: 200
Cost per month: 300
Employers: 20

Number of school: 179
Type of school: Kharkiv gymnasium
Pupils: 200
Cost per month: 300
Employers: 20

Menu:
1 - Add new objects
2 - Delete object
3 - Find by index
4 - Print all schools
5 - Print schools with employers more than average
6 - Print school with 2 or more words in name
0 - End program
Your choose:
```

Рисунок 6

ВИСНОВОК

В ході виконання лабораторної роботи навчився працювати з регулярними виразами та за допомогою регулярних виразів вдосконалив свою програму.