



**«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Компьютерные системы и сети

О т ч е т

по лабораторной работе № 2

Дисциплина: Современные средства разработки программного обеспечения

Название лабораторной работы: Реализация типовых решений с применением шаблонов проектирования и принципов SOLID

Студент гр. ИУ6-56

(Подпись, дата)

Т.А. Тищенко

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

М.В. Фетисов

(И.О. Фамилия)

Цель работы: Приобрести навыки применения принципов SOLID и шаблонов проектирования для решения конкретной задачи.

Задание: С использованием шаблона проектирования «декоратор» выполните создание объекта-декоратора карточки из вашего домашнего задания, не изменяя ее саму, добавив возможность вывода какого-либо признака (например, признака принадлежности этой карточки к особой коллекции, ее редкости, особой комбинации ее свойств и т.д.). Выполните вывод содержимого контейнера в стандартный поток с учетом выполненной декорации.

Задача: «Война в долине теней»

Карточка: «Воин Долины теней». Карточка должна содержать поля: фракция (перечисляемый тип), сила удара, защита, здоровье, ловкость, уклонение, тип боя (ближний, дальний).

```

#include <iostream>
#include <string>
#include <map>
#include <memory>

using namespace std;

class IOutput {
public:
    virtual void draw() = 0;
};

class WarriorVS: public IOutput {
    int _group, _power, _protection, _health, _skill, _deviation, _battle_type;
    string str;

public:
    WarriorVS(string s, int g, int p, int pr, int h, int sk, int d, int b) {
        str = s;
        _group = g;
        _power = p;
        _protection = pr;
        _health = h;
        _skill = sk;
        _deviation = d;
        _battle_type = b;
    }

    virtual void draw() override
    {
        cout << str << ": " << _group << ", " << _power << ", " << _protection << ", " << _health << _skill << ", "
            << _deviation << ", " << _battle_type << ", " << endl;
    }
};

class Decorator(IOutput *w) : wid(w) {}

    virtual void draw() override
    {
        wid->draw();
    }
};

class SpecialDecorator: public Decorator
{
public:
    SpecialDecorator(IOutput *w): Decorator(w){}

    virtual void draw() override
    {
        Decorator::draw();

        cout << "    Редкая коллекция" << endl;
    }
};

class ConnectDecorator: public Decorator
{
public:
    ConnectDecorator(IOutput *w): Decorator(w){}

    virtual void draw() override
    {
        Decorator::draw();

        cout << "    Вместе с Лоцаном и Матросами увеличивается защита и сила удара" << endl;
    }
};

int main()
{
    IOutput *aWidget = new ConnectDecorator(new SpecialDecorator(new WarriorVS("Капитан", 1, 4, 8, 7, 9, 4, 2)));
    aWidget->draw();
}

```

Листинг 1 – код программы

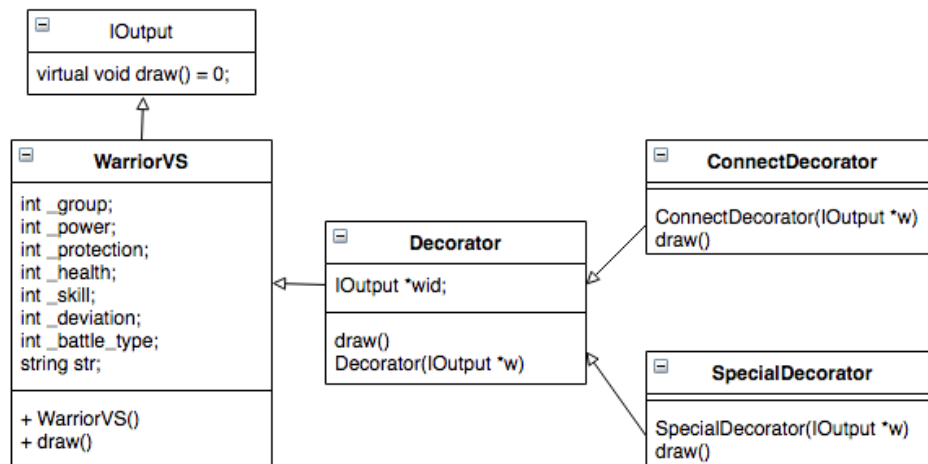


Рис.1 – диаграмма классов

Вывод программы:

Капитан: 1, 4, 8, 79, 4, 2,

Редкая коллекция

Вместе с Лоцаном и Матросами увеличивается защита и сила удара

Вывод: приобрел навыки применения принципов SOLID и шаблонов проектирования для решения конкретной задачи.