

# Тема 3: Наследование и виртуальность

## Цель

Получить навыки разработки собственных иерархий классов с использованием принципа расширения и виртуальности

## Общее задание

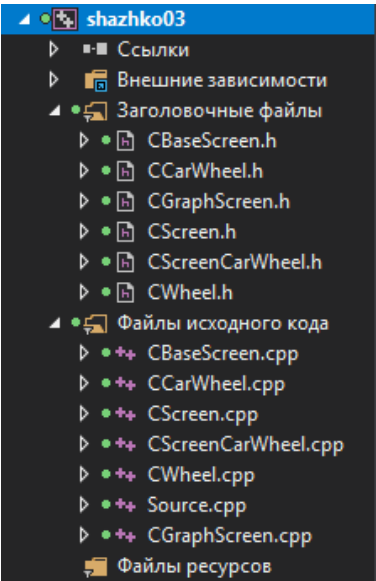
- Создать класс , используя наследование от с добавлением новых полей
- Выделить базовый класс для и с функцией BaseView::Display(): void и следующими виртуальными функциями, которые из нее вызываются: showHeader, showContent, showFooter
- Пренести основной функционал отображения в базовый класс, реализовав специфическое поведение в соответствующих виртуальных методах
- Создать класс , который задает специфику отображения для объектов
- Показать работу виртуальности на примере использования наследника через указатель на базовый класс

## Прикладная область







В соответствии с вариантом  
#17

Колеса	Колесо машины	Field1_data2 = назва гуми (рядок) Field2_data2 = виробник гуми (рядок)
--------	---------------	---

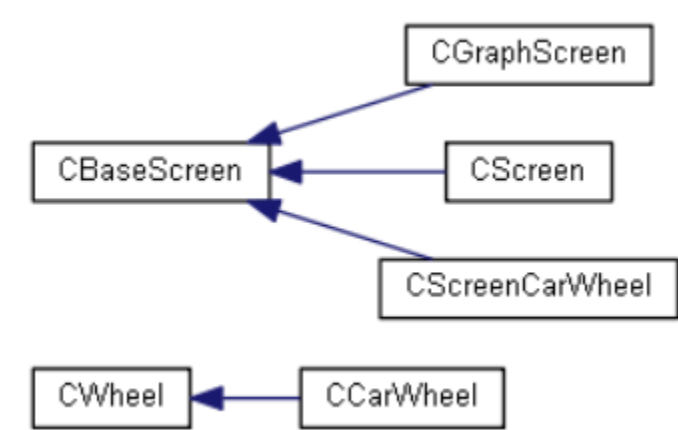
# Структура проекта



## Описание разработанных типов данных

 <b>CBaseScreen</b>	Класс описывающий базовое отображение объектов
 <b>CCarWheel</b>	Класс описывающий колесо машины
 <b>CGraphScreen</b>	Класс описывающий расширенное отображение объектов класса <b>CWheel</b>
 <b>CScreen</b>	Класс описывающий отображение объектов класса <b>CWheel</b>
 <b>CScreenCarWheel</b>	Класс описывающий отображение объектов класса <b>CCarWheel</b>
 <b>CWheel</b>	Класс описывающий колесо

## Диаграмма классов





## Разработанные методы и функции

---

См. раздел **members** на заголовочных страницах соответствующих типов данных

## Ссылки на файлы проекта

---

 <a href="#">CBaseScreen.cpp</a>	Содержит реализацию класса <b>CBaseScreen</b>
 <a href="#">CBaseScreen.h</a>	Содержит объявление базового класса <b>CBaseScreen</b>
 <a href="#">CCarWheel.cpp</a>	Содержит реализацию класса <b>CCarWheel</b>
 <a href="#">CCarWheel.h</a>	Содержит объявление базового класса <b>CCarWheel</b>
 <a href="#">CGraphScreen.h</a>	Содержит объявление класса <b>CGraphScreen</b>
 <a href="#">CScreen.cpp</a>	Содержит реализацию класса <b>CScreen</b>
 <a href="#">CScreen.h</a>	Содержит объявление класса <b>CScreen</b>
 <a href="#">CScreenCarWheel.cpp</a>	Содержит реализацию класса <b>CScreenCarWheel</b>
 <a href="#">CScreenCarWheel.h</a>	Содержит объявление класса <b>CScreenCarWheel</b>
 <a href="#">CWheel.cpp</a>	Содержит реализацию класса <b>CWheel</b>
 <a href="#">CWheel.h</a>	Содержит объявление класса <b>CWheel</b>
 <a href="#">Source.cpp</a>	

## Текст программы

---

```
int main() {
    CCarWheel *iWheel1 = new CCarWheel(6, 10, EUNITS_CENTIMETERS, "X10", "Canon");

    CScreen cscreen(iWheel1);
    cscreen.ShowHeader();
    cscreen.ShowContent();
    cscreen.ShowHeader();

    CGraphScreen graphScreen(iWheel1);
    graphScreen.ShowContent();
    std::cout << "\n";

    CScreenCarWheel screenCarWheel(iWheel1);
    screenCarWheel.ShowHeader();
    screenCarWheel.ShowContent();
    screenCarWheel.ShowHeader();

    delete iWheel1;
    return 0;
}
```

## Результаты работы

---

```
-----
      Diameter: 6
      Width: 10
      Units: CENTIMETERS
Volume: 282.743
-----
-----Graphics-----
****      |*****
*   *      |*****
*   *      |*****
*   *      |*****
*   *      |*****
*   *      |*****
****      |*****
-----
-----
      Diameter: 6
      Width: 10
      Units: CENTIMETERS
      Volume: 282.743
      Tire Manufacturer: Canon
      Tire Name: X10
-----
```

## Выводы

---

В ходе выполнения лабораторной работы были закреплены навыки разработки иерархий классов с использованием полиморфизма для задания отличного поведения объектов классов потомков от базовых классов с одинаковой сигнатурой определения