

Министерство образования Республики Беларусь
Учреждение образования «Белорусский государственный университет
информатики и радиоэлектроники»

Факультет компьютерного проектирования
Кафедра Информатики
Дисциплина «Программирование»

ОТЧЕТ

к лабораторной работе №6

на тему:

«НАСЛЕДОВАНИЕ.»

БГУИР 6-05-0612-02 17

Выполнил студент группы 353502
ХАРИТОНЧИК Денис Сергеевич

(дата, подпись студента)

Проверил ассистент каф.
Информатики
РОМАНЮК Максим Валерьевич

(дата, подпись преподавателя)

Минск 2024

1 ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ

Задание 1. Вариант 7. Выделить в предметной области 2-3 варианта сущности (например, "Геометрическая фигура", "Точка", "Треугольник"), отличающиеся несколькими полями и методами. Каждый класс имеет поля, свойства и методы. Спроектировать UML-диаграммы классов. Базовый класс для вашей иерархии объявите абстрактным. Он должен содержать абстрактные методы и методы с реализацией. Один из наследников должен перегружать метод родителя. Один из классов должен содержать виртуальный метод, который переопределяется в одном наследнике и не переопределяется в другом. Продемонстрировать работу всех объявленных методов. Продемонстрировать вызов конструктора родительского класса при наследовании. Продемонстрировать вызов метода родительского класса при его скрытии. Создать класс, закрытый для наследования. Примеры методов: вывод полей класса, изменение числовых полей класса (скорости, дохода, численности штата и т.д.), изменение текстовых полей класса (переименование), какая-либо динамика (полет, ремонт, переезд, открытие/закрытие и.д.). Предметная область: Компьютерная техника.

2 ВЫПОЛНЕНИЕ РАБОТЫ

В рамках проекта была создана директория ComputerEquipments, в которой размещаются 4 файла с классами ComputerEquipment, Laptop, Printer и ComputerPeripheral. Эти классы представляют основные компоненты для системы управления и демонстрации области компьютерной техники.

Класс ComputerEquipment является абстрактным базовым классом для компьютерной техники. Он содержит общие свойства и методы, которые присущи всей компьютерной технике. Класс имеет два свойства: Model для хранения модели устройства и Price для хранения его цены.

В классе определены два абстрактных метода: PowerOn() для включения устройства и PowerOff() для его выключения. Эти методы оставлены абстрактными, чтобы дочерние классы могли реализовать специфичное поведение для каждого типа компьютерной техники. Ниже приведен листинг кода класса ComputerEquipment.

```

public abstract class ComputerEquipment
{
    public string Model { get; set; }
    public decimal Price { get; set; }

    public abstract void PowerOn();

    public abstract void PowerOff();

    public abstract void GetPrice();

    public virtual void Upgrade()
    {
        Console.WriteLine($"Оборудование {Model} обновлено.");
    }
}

```

Класс Laptop представляет собой конкретный тип компьютерной техники - ноутбук. Он наследуется от абстрактного базового класса ComputerEquipment и расширяет его функциональность, добавляя специфичные для ноутбука свойства и методы. Класс имеет дополнительные свойства, такие как Processor, которое используется для хранения информации о процессоре ноутбука, и RAM, которое используется для хранения объема оперативной памяти. Кроме того, в нем содержится информация о размере экрана, типе графического процессора, объеме накопителя и наличии сканера отпечатков пальцев.

Класс Laptop реализует методы абстрактного класса ComputerEquipment, такие как PowerOn(), PowerOff() и GetPrice(), предоставляя свою собственную реализацию для каждого из них. Кроме того, он содержит дополнительный метод SetPassword(), который позволяет установить пароль для ноутбука, и метод GetInfoAboutLaptop(), который выводит информацию о ноутбуке, включая модель, процессор и объем оперативной памяти. Ниже приведен листинг кода класса Laptop.

```

public class Laptop : ComputerEquipment
{
    public string Processor { get; set; }
    public int RAM { get; set; }

    public Laptop(string model, decimal price, string processor, int ram)
    {
        Model = model;
        Price = price;
        Processor = processor;
        RAM = ram;
        Console.WriteLine($"Создан новый ноутбук {model}.");
    }

    public void SetPassword(string password)
    {
        Console.WriteLine($"Пароль для {Model} установлен: {password}");
    }
    public void GetInfoAboutLaptop()
    {
        Console.WriteLine($"У ноутбука {Model} процессор {Processor} и {RAM}
оперативной памяти");
    }

    public override void PowerOn()
    {
        Console.WriteLine($"Ноутбук {Model} включен и готов к использованию.");
    }

    public override void PowerOff()
    {
        Console.WriteLine($"Ноутбук {Model} выключен.");
    }
    public override void GetPrice()
    {
        Console.WriteLine($"Стоимость ноутбука ({Model}) - {Price}");
    }

    public override void Upgrade()
    {
        Console.WriteLine($"Ноутбук {Model} обновлен до новой версии.");
    }
}

```

Класс `Printer` представляет собой конкретный тип компьютерной техники - принтер. Он также наследуется от абстрактного базового класса `ComputerEquipment` и дополняет его функциональность, добавляя специфичные для принтера свойства и методы. Класс имеет дополнительное свойство `PrintType` для хранения информации о типе принтера (например, лазерный или струйный) и `PaperState` для хранения информации о наличии бумаги. Ниже приведен листинг кода класса `Printer`.

```

public class Printer : ComputerEquipment
{
    public string PrintType { get; set; }
    public bool PaperState { get; set; }

    public override void PowerOn()
    {
        Console.WriteLine($"{Model} включен.");
    }

    public override void PowerOff()
    {
        Console.WriteLine($"{Model} выключен.");
    }
    public override void GetPrice()
    {
        Console.WriteLine($"Стоимость принтера {Model} - {Price}");
    }

    public void PrintDocument(string document)
    {
        if (PaperState)
        {
            Console.WriteLine($"Документ {document} успешно напечатан на
принтере {Model}");
        }
        else
        {
            Console.WriteLine($"Ошибка: Нет бумаги на принтере {Model}.");
        }
    }
    public void GetStateAndInfo()
    {
        Console.WriteLine($"У принтера {Model} состояние бумаги - {PaperState}
и тип его печати {PrintType}");
    }

    public void ReplaceCartridge()
    {
        Console.WriteLine($"Картридж на принтере {Model} заменен.");
    }
}

```

Класс `ComputerPeripheral` представляет собой конкретный тип компьютерной техники - периферийное устройство. Он также наследуется от абстрактного базового класса `ComputerEquipment` и дополняет его функциональность, добавляя специфичные для периферийного устройства свойства и методы.

Класс имеет дополнительное свойство для хранения модели устройства. Кроме того, в классе определены конструктор и переопределены

методы `PowerOn()` и `PowerOff()`, чтобы реализовать функциональность включения и выключения периферийного устройства. Ниже приведен листинг кода класса `ComputerPeripheral`.

```
public sealed class ComputerPeripheral : ComputerEquipment
{
    public string Model { get; set; }
    public decimal Price { get; set; }

    public ComputerPeripheral(string model, decimal price)
    {
        Model = model;
        Price = price;
        Console.WriteLine($"Создано новое периферийное компьютерное устройство
{model}.");
    }
    public override void PowerOn()
    {
        Console.WriteLine($"{Model} включен");
    }
    public override void PowerOff()
    {
        Console.WriteLine($"{Model} выключен.");
    }
    public override void GetPrice()
    {
        Console.WriteLine($"Стоимость {Model} - {Price}");
    }

    public void Connect()
    {
        Console.WriteLine($"{Model} подключен.");
    }
}
```

Класс `Program` представляет точку входа в приложение. Внутри метода `Main` создаются объекты классов `Laptop`, `Printer` и `ComputerPeripheral`, которые управляют работой различных компонентов компьютерной техники. Для начала инициализируется объект класса `Laptop` с помощью конструктора, передавая данные о модели, цене, процессоре и объеме оперативной памяти. Затем вызываются различные методы этого объекта, такие как `PowerOn`, `GetPrice`, `GetInfoAboutLaptop`, `SetPassword`, `Upgrade` и `PowerOff`, для демонстрации его функциональности.

Далее инициализируется объект класса `Printer`, передавая данные о модели, цене, типе печати и наличии сканера. Затем вызываются методы для включения и выключения принтера (`PowerOn` и `PowerOff`), получения информации о его состоянии, печати документа и замены картриджа.

Наконец, создается объект класса `ComputerPeripheral`, передавая данные о модели и цене. Затем вызываются методы для подключения устройства, включения и выключения.

Этот класс обеспечивает демонстрацию работы различных компонентов компьютерной техники, их функциональности и взаимодействия. На рисунке 1 изображен вывод данных в консоль. На рисунке 2 изображена UML-диаграмма классов. Ниже приведен листинг кода класса `Program`.

```
class Program
{
    static void Main(string[] args)
    {
        Laptop laptop = new Laptop("MacBook Air M1 2020", 3100, "M1", 8);

        laptop.PowerOn();
        laptop.GetPrice();
        laptop.GetInfoAboutLaptop();
        laptop.SetPassword("qwerty");
        laptop.Upgrade();
        laptop.PowerOff();
        Console.WriteLine("-----");

        Printer printer = new Printer( "HP LaserJet Pro MFP M28w", 300,
"лазерный", true);

        printer.PowerOn();
        printer.GetPrice();
        printer.GetStateAndInfo();
        printer.PrintDocument("Отчет");
        printer.ReplaceCartridge();
        printer.PowerOff();
        Console.WriteLine("-----");

        ComputerPeripheral computerPeripheral = new
ComputerPeripheral("Keychron V3", 350);

        computerPeripheral.PowerOn();
        computerPeripheral.GetPrice();
        computerPeripheral.Connect();
        computerPeripheral.PowerOff();
    }
}
```

```

-----
Создан новый принтер HP LaserJet Pro MFP M28w.
Принтер HP LaserJet Pro MFP M28w включен в помещении офис.
Стоимость принтера HP LaserJet Pro MFP M28w - 300
У принтера HP LaserJet Pro MFP M28w состояние бумаги - True и тип его печати лазерный
Документ Отчет успешно напечатан на принтере HP LaserJet Pro MFP M28w
Картридж на принтере HP LaserJet Pro MFP M28w заменен.
HP LaserJet Pro MFP M28w выключен.
-----

Создано новое периферийное компьютерное устройство Keychron V3.
Keychron V3 включен
Стоимость Keychron V3 - 350
Keychron V3 подключен.
Keychron V3 выключен.

```

Рисунок 1 – Работа консольного приложения



Рисунок 2 – UML-диаграмма классов

ВЫВОД

В ходе выполнения лабораторной работы были изучены основные принципы наследования классов и их особенности. Были освоены навыки реализации UML-диаграмм классов. Отточена и закреплена работа с классами и разделением сущностей. Изучен метод для закрытия наследования классов.