**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**КАФЕДРА САПР**

**ОТЧЕТ**

**по лабораторной работе №9**

**по дисциплине «Объектно-ориентированное программирование»**

|  |  |  |
| --- | --- | --- |
| Студенты гр. 9301 |  | Примакова Е.Е.. |
| Преподаватель |  | Новакова Н.Е. |

Санкт-Петербург

2021

# Цель работы

Получить навыки практической работы с классами в языке C#. Закрепить изученный за семестр теоретический материал по теме. Самостоятельно разработать систему классов.

# Анализ задачи

Необходимо реализовать систему классов для представления склада продуктов

# Теоритическая часть

Абстрактные классы — классы, реализованные с ключевым словом abstract. Ключевая особенность этих классов состоит в том, что создавать экземпляры этих классов нельзя. Данные классы используются для описания общих свойств некоторых классов объектов, после чего другие классы их наследуют. Таким образом, у классов наследуются поля и методы, а при необходимости что-то поменять меняется только абстрактный класс, а не все классы, наследовавшие его, по отдельности.

Наследование классов. Благодаря наследованию один класс может унаследовать функциональность другого класса. Все классы по умолчанию могут наследоваться. Однако здесь есть ряд ограничений: не поддерживается множественное наследование — класс может наследоваться только от одного класса; при создании производного класса надо учитывать тип доступа к базовому классу — тип доступа к производному классу должен быть таким же, как и у базового класса, или более строгим. То есть, если базовый класс у нас имеет тип доступа internal, то производный класс может иметь тип доступа internal или private, но не public, однако следует также учитывать, что если базовый и производный класс находятся в разных сборках (проектах), то в этом случае производный класс может наследовать только от класса, который имеет модификатор public; если класс объявлен с модификатором sealed, то от этого класса нельзя наследовать и создавать производные классы; нельзя наследовать от статического класса.

# Формальная постановка задачи

## Исходные данные

## Результирующие (выходные) данные

# Текст программы

# Контрольный пример

# Полученные результаты

# Выводы

В ходе выполнения данной лабораторной работы были закреплены навыки работы с классами в C# и на их основе была реализована модель, описывающая склад продуктов.