

Практическая часть к **Уроку №8 Наследование и полиморфизм.**

№	Задание
1.	Используя Visual Studio, создайте проект по шаблону Console Application. Создайте программу, в которой создайте класс хвост, который содержит в себе поля длину хвоста типа <code>int</code> и вид хвоста типа <code>string</code> , создать свойства доступа и конструктор пользовательский класса. Создать класс хвостатое животное, содержащий хвост, цвет(строка), возраст. Определить <code>public</code> - производный класс собака, имеющий дополнительный параметр-кликку (строка). В классе собака создать метод для отображения полной информации о собаке.
2.	Используя Visual Studio, создайте проект по шаблону Console Application. Создайте программу, в которой создайте базовый класс Shape (фигура), который содержит в себе поле типа <code>double</code> с именем Volume и метод типа <code>double</code> GetVolume() который должен вернуть объём фигуры. Далее создайте классы: Pyramid (пирамида), Cylinder (цилиндр), Ball (шар), которые будут наследоваться от класса Shape, реализуйте в каждом из них метод для нахождения объёма. Создайте класс Box (ящик) – он является контейнером, может содержать в себе другие фигуры. В классе Box создайте поле типа <code>double</code> с именем DrawerVolume (Объём ящика), поле должно хранить в себе объём ящика. Далее в классе Box создайте метод Add(), который принимает на вход объекты типа Shape, и возвращает значение типа <code>bool</code> . В классе Box реализуйте логику для добавления новых фигуры до тех пор, пока для них хватает места в Box (будем считать только объём, игнорируя форму, например, мы переливаем жидкость). Если места для добавления новой фигуры не хватает, то метод должен вернуть <code>false</code> .
3.	Используя Visual Studio, создайте проект по шаблону Console Application. Требуется: Создать класс, представляющий учебный класс Classroom. Создайте класс ученик Pupil. В теле класса создайте методы <code>void Study()</code> , <code>void Read()</code> , <code>void Write()</code> , <code>void Relax()</code> . Создайте 3 производных класса ExcelentPupil, GoodPupil, BadPupil от класса базового класса Pupil и переопределите каждый из методов, в зависимости от успеваемости ученика. Конструктор класса Classroom принимает аргументы типа Pupil, класс должен состоять из 4 учеников. Предусмотрите возможность того, что пользователь может передать 2 или 3 аргумента. Выведите информацию о том, как все ученики экземпляра класса Classroom умеют учиться, читать, писать, отдыхать.
4.	Используя Visual Studio, создайте проект по шаблону Console Application. Создайте программу, в которой создайте базовый класс <b>Person</b> (человек), в классе создайте поле типа <code>int</code> с именем BirthYear (год рождения), поле типа <code>string</code> с именем Name и поле типа <code>string</code> с именем Surname. Далее создайте классы <b>Student</b> (студент), <b>Teacher</b> (преподаватель). В классе Student добавьте поле типа <code>string[]</code> с именем Study Courses (изучаемые курсы), свойство (не авто свойство) для добавления (set) и получения (get) изучаемых курсов и метод DisplayStudyCourses() с возвращаемым значением <code>void</code> который будет выводить на консоль все предметы, максимальное количество изучаемых курсов = 3. В классе преподаватель создать поле типа <code>Student[]</code> с именем StudentsArray, и свойство (не авто свойство) для добавления (set) и получения (get) студентов. Создайте 5 экземпляров класса типа <b>Student</b> и инициализируйте их произвольными значениями, и 2 экземпляра класса типа <b>Teacher</b> , инициализируйте их произвольными значениями (для инициализации поля StudentsArray используйте уже созданные экземпляры <b>Student</b> ). Далее создайте класс PeopleInfo, в нем создайте поле типа <code>Person[]</code> с именем PeopleArray и свойство (не авто свойство) для добавления (set) и получения (get) людей и метод который будет выводить всех людей который есть в поле PeopleInfo в порядке возрастания возраста.

**Проверочные задачи к уроку №8.**

1	<p>Используя Visual Studio, создайте проект по шаблону Console Application.</p> <p>Создайте программу, в которой создайте класс Car (машина) который содержит в себе полями с название, цвет, цена, const полем CompanyName. Создать два конструктора – по умолчанию и с параметрами. Создать свойство доступа к полю цвет. Определить методы Input () - для ввода данных о машине с консоли , Print () - для вывода данных о машине на консоль и ChangePrice (double x) - для изменения цены на x%. Ввести данные о 3 авто. Уменьшить их цену на 10%, вывести данные об авто. Ввести новый цвет и покрасить авто с цветом white в указанный цвет.</p>
---	--