

Федеральное государственное бюджетное образовательное учреждение
высшего образования
Югорский государственный университет
Институт цифровой экономики
(ИЦЭ)

ОТЧЕТ
по лабораторной работе №3
Тема:
«Наследование»
По дисциплине:
«Языки программирования высокого уровня»

Отчет выполнила:
студентка группы 1182б
Гаджиева Айсель Лазим кызы
Отчет проверил: доцент
Сафонов Егор Иванович

г. Ханты-Мансийск – 2019 г.

Цель работы:

Написание классов с методами, и их наследование.

Постановка задачи:

1. Написать два класса: Horse(лошадь) - в конструкторе задается кличка лошади и Pegasus(пегас) - реализующий метод полет(сообщает какая лошадь летит). Унаследовать пегаса от лошади.
2. Написать три класса: Pet(домашнее животное) - с полями вес, пол, возраст, Cat(кот) с полем имя и Dog(собака) с полем имя. Унаследовать кота и собаку от животного. В классах кот и собака реализовать методы вывода информации и о животных
3. Написать четыре класса: Fish (Рыбы), Animal(Животные), Ape(Обезьяны), Human(Человек). Унаследовать животных от рыб, обезьян от животных и человека от обезьян. Поля и методы разработать самостоятельно.
4. Скрыть все внутренние переменные класса Cat, добавить методы работы с этими переменными

Листинг кода A1 – Создание классов и реализация методов. Наследование классов:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace ConsoleApp1
{
    class Program
    {
        static void Main(string[] args)
        {
            string name;
            string Name;
            Console.WriteLine("Введите кличку питомца (Cat)");
            name = Console.ReadLine();
            Console.WriteLine("Введите кличку питомца (Dog)");
            Name = Console.ReadLine();
            Cat cat = new Cat(name);
            Dog dog = new Dog(Name);
            Console.WriteLine("Информация о Cat");
            cat.Info();
            cat.Age = 4;
            cat.Weight = 10;
            Console.WriteLine("\nИнформация о Dog");
            dog.Info();
            Fish fish = new Fish();
            fish.Info();
            Animal animal = new Animal();
            animal.Info();
        }
    }
}
```

```

        Ape ape = new Ape();
        ape.Info();
        Human human = new Human();
        human.Info();
        Console.ReadKey();
    }
}
class Horse
{
    protected string name;
    public Horse()
    {
        Console.WriteLine("Введите кличку лошадки");
        name = Console.ReadLine();
    }
}
class Pegas : Horse
{
    public Pegas()
    {
        Console.WriteLine(name + " эволюционировала в
пегаса!!!");
    }
    public void Fly()
    {
        Console.WriteLine(name + " может летать");
    }
}
class Pet
{
    protected short age;
    protected float weight;
    protected bool gender;
    public Pet()
    {
        Console.WriteLine("Введите возраст");
    }
}

```

```

        while (!short.TryParse(Console.ReadLine(), out
age)||age<0) ;
        Console.WriteLine("Введите вес");
        while (!float.TryParse(Console.ReadLine(), out
weight)||weight<0.3) ;
        Console.WriteLine("Введите пол [м-мужской; ж-
женский]");
        if (Console.Read() == 'м')
        { gender = true; }
        else
        { gender = false; }
    }
}
class Cat : Pet
{
    protected string name;
    public Cat(string name)
    {
        this.name = name;
    }
    public void Info()
    {
        Console.WriteLine("Перед вами " + name + ": " );
        if(gender)
        {
            if (weight>=7) { Console.Write("с лишним весом
"); }
            else if (weight<1&&age>0) {
Console.Write("тощий "); }
            if (age>=7) { Console.Write("старый кот"); }
            else if (age<7&&age>0) {
Console.Write("молодой кот"); }
            else { Console.Write("котёнок-мальчик."); }
            Console.WriteLine();
        }
        else

```

```

        {
            if (weight >= 7) { Console.WriteLine("с лишним
весом "); }

            else if (weight<1 && age>0) {
Console.WriteLine("тощая "); }

            if (age>=7) { Console.WriteLine("старая кошка"); }
            else if (age<7 && age > 0) {
Console.WriteLine("молодая кошка"); }
            else { Console.WriteLine("котёнок-девочка"); }
            Console.WriteLine();
        }
    }

    public short Age { get { return age; } set { if (value
> age) age = value; } }

    public float Weight { get { return weight; } set {
weight = value; } }

    public char Gender { get { if (gender) return 'м';
else return 'ж'; } }
}

class Dog : Pet
{
    protected string name;
    public Dog(string name)
    {
        this.name = name;
    }

    public void Info()
    {
        Console.WriteLine("Перед вами " + name + ": ");
        if (gender)
        {
            if (weight>32) { Console.WriteLine("с лишним весом
"); }

            else if (weight<15 && age>0) {
Console.WriteLine("тощая мужская особь"); }

```

```

        if (age >= 10) { Console.Write("старая
мужская особь"); }
        else if (age < 10 && age > 0) {
Console.Write("молодая мужская особь"); }
        else { Console.Write("щенок-мальчик"); }
        Console.WriteLine();
    }
    else
    {
        if (weight>27) { Console.Write("с лишним весом
"); }
        else if (weight<11 && age>0) {
Console.Write("тощая женская особь"); }
        if (age >= 10) { Console.WriteLine("старая
женская особь"); }
        else if (age < 10 && age>0) {
Console.WriteLine("молодая женская особь"); }
        else { Console.Write("щенок-девочка"); }
        Console.WriteLine();
    }
}
}
class Fish
{
    protected string breathe;//дышать
    protected string name;//имя
    private short fin;//плавник
    protected bool tail;//хвост
    private short gills;//жабры
    private bool squama;//чешуя
    public Fish()
    {
        Console.WriteLine("Введите имя:");
        name = Console.ReadLine();
    }
    public void Info()

```

```

        {
            breathe = "Рыбы дышат растворённым в воде
кислородом";
            Console.WriteLine("Введите число плавников:");
            while (!short.TryParse(Console.ReadLine(), out
fin) || fin > 15) ;
            Console.WriteLine("Количество жаберных дуг:");
            while (!short.TryParse(Console.ReadLine(), out
gills) || gills > 10) ;
            Console.WriteLine("Чешуйчатая рыба? [д]-да, [всё
остальное]-нет");
            if (Console.Read() == 'д')
            { squama = true; }
            else
            { squama = false; }
            Console.WriteLine("Перед вами " + name + ": ");
            if (squama)
            {
                Console.Write($"Чешуйчатая рыба с: {fin}
плавником(-ми), хвостом, количеством жаберных дуг = {gills},
({breathe}!)");
            }
            else
            {
                Console.Write($"Рыба без чешуи с: {fin}
плавниками(-ми), хвостом, количеством жаберных дуг = {gills},
({breathe}!)");
            }
        }
    }

    class Animal : Fish
    {
        protected string walk;
        protected string wool;
        protected string spine;
        public new void Info()
    }
}

```



```

        {
            Console.WriteLine("Животное имеет шерсть? Если да,
то опишите его");
            wool = Console.ReadLine();
            spine = "Позвоночное животное";
            walk = "ходящее на 4 лапах";
            breathe = "Животное дышит кислородом";
            Console.WriteLine("Животное имеет хвост? [д]-да,
[всё остальное]-нет");
            if (Console.Read() == 'д')
            { tail = true; }
            else
            { tail = false; }
            Console.WriteLine("Перед вами " + name + ": ");
            if (tail)
            {
                Console.Write($"Животное      с      хвостом,
{walk}, описание шерсти: {wool} , ({spine},{breathe}!)");
            }
            else
            {
                Console.Write($"Животное      без      хвоста,
{walk}, описание шерсти: {wool} , ({spine},{breathe}!)");
            }
        }
    }

    class Ape : Animal
    {
        private string tail1;
        public new void Info()
        {
            breathe = "Животные дышат кислородом";
            wool = "Тело обезьян покрыто шерстью разной
окраски, в зависимости от вида она может быть светло-
коричневой, рыжей, черно - белой, серо - оливковой.";
        }
    }
}

```

tail1 = "У некоторых обезьян в наличие очень длинный хвост, длина которого может даже превышать размеры тела, хвост обезьяны выполняет функцию балансира при перемещении между деревьями. Зато у обезьян, живущих на земле хвост совсем короткий. Что же касается обезьян без хвоста, то его не имеют все «человекоподобные» обезьяны (как впрочем, не имеют его и люди).";

```
tail = true;
```

```
Console.WriteLine($"Перед вами {name}:\nЖивотное  
с хвостом : {tail1} ; {walk} ; описание шерсти: {wool} ;  
({spine},{breathe}!) \n");
```

```
}
```

```
}
```

```
class Human : Ape
```

```
{
```

```
public new void Info()
```

```
{
```

```
    spine = "Прямой позвоночник с прогибом в  
пояснице";
```

```
    breathe = "Дышат кислородом на земле";
```

```
    wool = "шерсть отсутствует";
```

```
    tail = false;
```

```
    walk = "прямохождение (ходящее на 2 ногах)";
```

```
    Console.WriteLine($"Перед вами {name}:\nЧеловек!  
: {walk} ; описание шерсти: {wool} ; ({spine},{breathe}!)  
");
```

```
}
```

```
}
```

```
}
```

Заключение:

В ходе работы была достигнута цель.