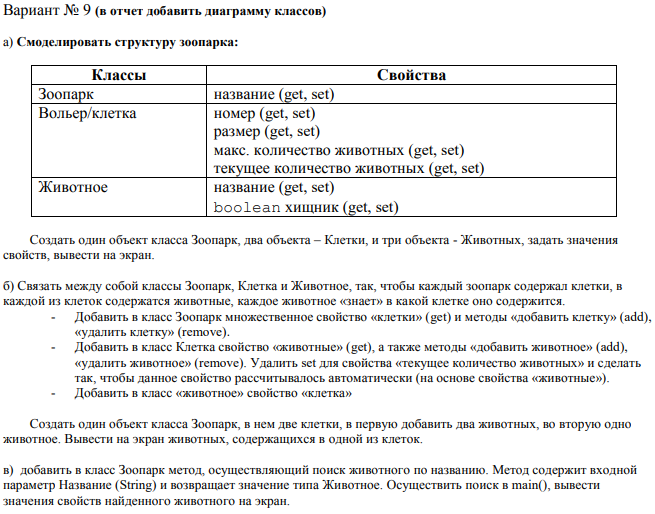
**ЛАБОРАТОРНАЯ РАБОТА №7**

**СОЗДАНИЕ КЛАССОВ И РАБОТА С ОБЪЕКТАМИ**

**Цель работы:** получить основные понятия и навыки по созданию классов, описанию свойств и работы с объектами.



**A)**

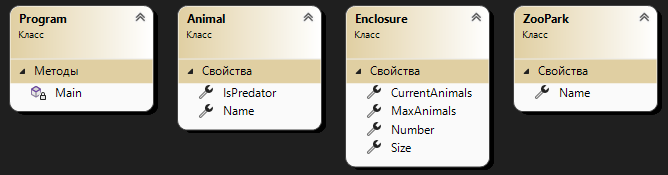


Рисунок 1 Диаграмма классов

**Листинг:**

using System;

class ZooPark

{

public string Name { get; set; }

}

class Enclosure

{

public int Number { get; set; }

public string Size { get; set; }

public int MaxAnimals { get; set; }

public int CurrentAnimals { get; set; }

}

class Animal

{

public string Name { get; set; }

public bool IsPredator { get; set; }

}

class Program

{

static void Main(string[] args)

{

// Создаем зоопарк

ZooPark zoo = new ZooPark();

zoo.Name = "Центральный зоопарк";

// Создаем два вольера

Enclosure enclosure1 = new Enclosure();

enclosure1.Number = 1;

enclosure1.Size = "Средний";

enclosure1.MaxAnimals = 5;

enclosure1.CurrentAnimals = 2;

Enclosure enclosure2 = new Enclosure();

enclosure2.Number = 2;

enclosure2.Size = "Большой";

enclosure2.MaxAnimals = 10;

enclosure2.CurrentAnimals = 3;

// Создаем три животных

Animal animal1 = new Animal();

animal1.Name = "Лев";

animal1.IsPredator = true;

Animal animal2 = new Animal();

animal2.Name = "Зебра";

animal2.IsPredator = false;

Animal animal3 = new Animal();

animal3.Name = "Тигр";

animal3.IsPredator = true;

// Выводим информацию на экран

Console.WriteLine("Информация о зоопарке:");

Console.WriteLine("Название: " + zoo.Name);

Console.WriteLine("\nИнформация о вольерах:");

Console.WriteLine("Вольер 1:");

Console.WriteLine("Номер: " + enclosure1.Number);

Console.WriteLine("Размер: " + enclosure1.Size);

Console.WriteLine("Максимальное количество животных: " + enclosure1.MaxAnimals);

Console.WriteLine("Текущее количество животных: " + enclosure1.CurrentAnimals);

Console.WriteLine("\nВольер 2:");

Console.WriteLine("Номер: " + enclosure2.Number);

Console.WriteLine("Размер: " + enclosure2.Size);

Console.WriteLine("Максимальное количество животных: " + enclosure2.MaxAnimals);

Console.WriteLine("Текущее количество животных: " + enclosure2.CurrentAnimals);

Console.WriteLine("\nИнформация о животных:");

Console.WriteLine("Животное 1:");

Console.WriteLine("Название: " + animal1.Name);

Console.WriteLine("Хищник: " + (animal1.IsPredator ? "Да" : "Нет"));

Console.WriteLine("\nЖивотное 2:");

Console.WriteLine("Название: " + animal2.Name);

Console.WriteLine("Хищник: " + (animal2.IsPredator ? "Да" : "Нет"));

Console.WriteLine("\nЖивотное 3:");

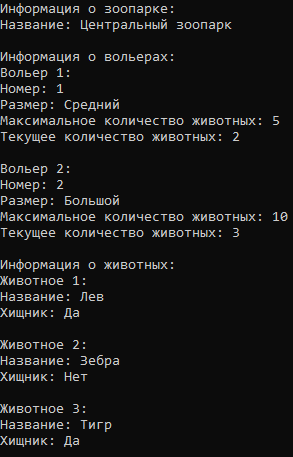
Console.WriteLine("Название: " + animal3.Name);

Console.WriteLine("Хищник: " + (animal3.IsPredator ? "Да" : "Нет"));

Console.ReadLine();

}

}



Б)

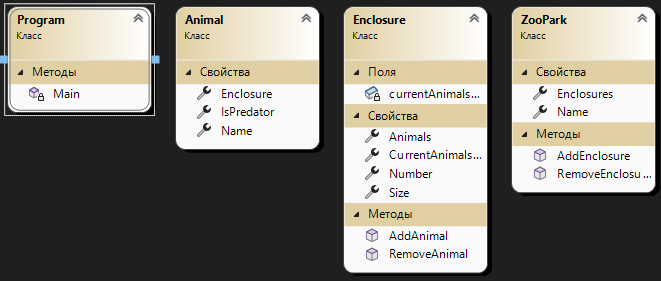


Рисунок 2 диаграмма классов

Листинг

using System;

using System.Collections.Generic;

class ZooPark

{

public string Name { get; set; }

public List<Enclosure> Enclosures { get; } = new List<Enclosure>();

public void AddEnclosure(Enclosure enclosure)

{

Enclosures.Add(enclosure);

}

public void RemoveEnclosure(Enclosure enclosure)

{

Enclosures.Remove(enclosure);

}

}

class Enclosure

{

public int Number { get; set; }

public string Size { get; set; }

public List<Animal> Animals { get; } = new List<Animal>();

private int currentAnimalsCount;

public int CurrentAnimalsCount

{

get

{

return currentAnimalsCount;

}

}

public void AddAnimal(Animal animal)

{

Animals.Add(animal);

currentAnimalsCount = Animals.Count;

}

public void RemoveAnimal(Animal animal)

{

if (Animals.Remove(animal))

{

currentAnimalsCount = Animals.Count;

}

}

}

class Animal

{

public string Name { get; set; }

public bool IsPredator { get; set; }

public Enclosure Enclosure { get; set; }

}

class Program

{

static void Main(string[] args)

{

// Создаем зоопарк

ZooPark zoo = new ZooPark();

zoo.Name = "Центральный зоопарк";

// Создаем две клетки

Enclosure enclosure1 = new Enclosure();

enclosure1.Number = 1;

enclosure1.Size = "Средний";

zoo.AddEnclosure(enclosure1);

Enclosure enclosure2 = new Enclosure();

enclosure2.Number = 2;

enclosure2.Size = "Большой";

zoo.AddEnclosure(enclosure2);

// Создаем три животных и добавляем их в клетки

Animal animal1 = new Animal();

animal1.Name = "Лев";

animal1.IsPredator = true;

animal1.Enclosure = enclosure1;

enclosure1.AddAnimal(animal1);

Animal animal2 = new Animal();

animal2.Name = "Зебра";

animal2.IsPredator = false;

animal2.Enclosure = enclosure1;

enclosure1.AddAnimal(animal2);

Animal animal3 = new Animal();

animal3.Name = "Тигр";

animal3.IsPredator = true;

animal3.Enclosure = enclosure2;

enclosure2.AddAnimal(animal3);

// Выводим на экран животных в одной из клеток

Console.WriteLine("Животные в первой клетке:");

foreach (var animal in enclosure1.Animals)

{

Console.WriteLine(animal.Name);

}

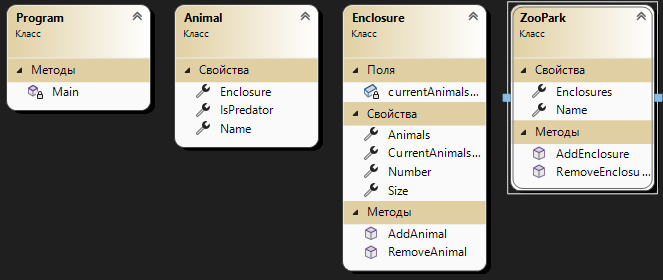
Console.ReadLine();

}

}



В)



Листинг

using System;

using System.Collections.Generic;

class ZooPark

{

public string Name { get; set; }

public List<Enclosure> Enclosures { get; } = new List<Enclosure>();

public void AddEnclosure(Enclosure enclosure)

{

Enclosures.Add(enclosure);

}

public void RemoveEnclosure(Enclosure enclosure)

{

Enclosures.Remove(enclosure);

}

public Animal FindAnimalByName(string name)

{

foreach (var enclosure in Enclosures)

{

foreach (var animal in enclosure.Animals)

{

if (animal.Name == name)

{

return animal;

}

}

}

return null; // Возвращаем null, если животное не найдено

}

}

class Enclosure

{

public int Number { get; set; }

public string Size { get; set; }

public List<Animal> Animals { get; } = new List<Animal>();

private int currentAnimalsCount;

public int CurrentAnimalsCount

{

get

{

return currentAnimalsCount;

}

}

public void AddAnimal(Animal animal)

{

Animals.Add(animal);

currentAnimalsCount = Animals.Count;

}

public void RemoveAnimal(Animal animal)

{

if (Animals.Remove(animal))

{

currentAnimalsCount = Animals.Count;

}

}

}

class Animal

{

public string Name { get; set; }

public bool IsPredator { get; set; }

public Enclosure Enclosure { get; set; }

}

class Program

{

static void Main(string[] args)

{

// Создаем зоопарк

ZooPark zoo = new ZooPark();

zoo.Name = "Центральный зоопарк";

// Создаем две клетки

Enclosure enclosure1 = new Enclosure();

enclosure1.Number = 1;

enclosure1.Size = "Средний";

zoo.AddEnclosure(enclosure1);

Enclosure enclosure2 = new Enclosure();

enclosure2.Number = 2;

enclosure2.Size = "Большой";

zoo.AddEnclosure(enclosure2);

// Создаем три животных и добавляем их в клетки

Animal animal1 = new Animal();

animal1.Name = "Лев";

animal1.IsPredator = true;

animal1.Enclosure = enclosure1;

enclosure1.AddAnimal(animal1);

Animal animal2 = new Animal();

animal2.Name = "Зебра";

animal2.IsPredator = false;

animal2.Enclosure = enclosure1;

enclosure1.AddAnimal(animal2);

Animal animal3 = new Animal();

animal3.Name = "Тигр";

animal3.IsPredator = true;

animal3.Enclosure = enclosure2;

enclosure2.AddAnimal(animal3);

// Выводим на экран животных в одной из клеток

Console.WriteLine("Животные в первой клетке:");

foreach (var animal in enclosure1.Animals)

{

Console.WriteLine(animal.Name);

}

Console.ReadLine();

// Осуществляем поиск животного по названию

string searchName = "Лев";

Animal foundAnimal = zoo.FindAnimalByName(searchName);

// Выводим информацию о найденном животном

if (foundAnimal != null)

{

Console.WriteLine($"Найдено животное с названием '{searchName}':");

Console.WriteLine($"Имя: {foundAnimal.Name}");

Console.WriteLine($"Хищник: {(foundAnimal.IsPredator ? "Да" : "Нет")}");

Console.WriteLine($"Клетка: {foundAnimal.Enclosure.Number}");

}

else

{

Console.WriteLine($"Животное с названием '{searchName}' не найдено.");

}

Console.ReadLine();

}

}

