**21 Паттерны проектирования**

Задания №1. Реализовать паттерны по примерам из теоретического материала согласно вариантам. Вариант 1.

Листинг программы:

using DuckLibrary;

namespace DuckLibrary

{

public abstract class Duck

{

protected IFlyBehavior flyBehavior;

protected IQuackBehavior quackBehavior;

public Duck() { }

public abstract void Display();

public void PerformFly() => flyBehavior?.Fly() ?? Console.WriteLine("Эта утка не умеет летать.");

public void PerformQuack() => quackBehavior?.Quack() ??

Console.WriteLine("Эта утка молчит.");

public void Swim() => Console.WriteLine("Все утки плавают, даже приманки!");

public void SetFlyBehavior(IFlyBehavior fb) => flyBehavior = fb;

public voidSetQuackBehavior(IQuackBehavior qb) => quackBehavior = qb;

}

}

namespace DuckLibrary

{

public interface IFlyBehavior

{

void Fly();

}

}

namespace DuckLibrary

{

public interface IQuackBehavior

{

void Quack();

}

}

namespace DuckLibrary

{

public class FlyWithWings : IFlyBehavior

{

public void Fly() =>

Console.WriteLine("Я летаю с крыльями!");

}

}

namespace DuckLibrary

{

public class NoFly : IFlyBehavior

{

public void Fly() => Console.WriteLine("Я не умею летать.");

}

}

namespaceDuckLibrary

{

public class Quack : IQuackBehavior

{

public void Quack() =>

Console.WriteLine("Кря-кря!");

}

}

namespaceDuckLibrary

{

public class MuteQuack : IQuackBehavior

{

public void Quack() =>

Console.WriteLine("...");

}

}

Анализ результатов:

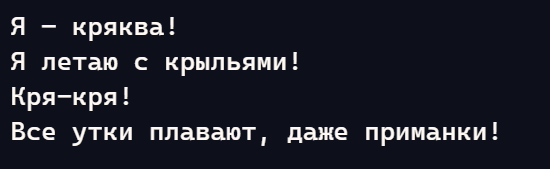


Рисунок 21.1 – Результаты работы программы