

1.

**В результате выполнения фрагмента программы:**

```
using System;
using System.Linq;

class Program {
    static void Main() {
        var args = new int[] { 0, 1, 2, 3, 4, 5, 6 };
        var res = from t in args
                   where t >= 3 && t <= 5
                   select t % 4;
        foreach (var item in res) {
            Console.Write(item);
        }
    }
}
```

**на экран будет выведено:**

2.

**Ключевыми словами языка LINQ являются (укажите все верные ответы):**

- 1) **from.**
- 2) **select.**
- 3) **yield.**
- 4) **async.**
- 5) **await.**

3.

**В результате выполнения фрагмента программы:**

```
using System;
using System.Linq;

public class Integers1 {
    public int Hash;
    public int val;

    public Integers1(int x) {
        val = x;
        Hash = x ^ 2;
    }
}

public class Integers2 {
    public int Hash;
    public int val;

    public Integers2(int x) {
        val = x;
        Hash = x ^ 3;
    }
}

class Program {
    static void Main() {
        Integers1[] args1 = new Integers1[2];
        Integers2[] args2 = new Integers2[2];
        for (int i = 0; i < 2; i++) {
            args1[i] = new Integers1(i);
            args2[i] = new Integers2(i * 2);
        }
        var query = from a in args1
                     join b in args2 on a.Hash equals b.Hash
                     select a.val;
        foreach (var q in query)
            Console.Write(q);
    }
}
```

**на экран будет выведено:**

4.

**В результате выполнения фрагмента программы:**

```
using System;
using System.Linq;

class Program {
    static void Main() {
        var A = new[] { 1, 2 };
        var B = new[] { 5, 7 };
        var someInts = from a in A
                        from b in B
                        let mult = a + b
                        where mult > 5
                        select a + b;
        foreach (var a in someInts)
            Console.Write(a);
    }
}
```

на экран будет выведено:

5.

**В результате выполнения фрагмента программы:**

```
using System;
using System.Linq;

class Program {
    static void Main() {
        var A = new[] { 30, 26 };
        var B = new[] { 4, 7, 3 };
        var someInts = from a in A
                        from b in B
                        let mod = a % b
                        orderby mod
                        where mod > 1
                        select mod;
        foreach (var a in someInts)
            Console.Write(a);
    }
}
```

на экран будет выведено:

6.

**В результате выполнения фрагмента программы:**

```
using System;

class Program {
    static void Main() {
        var integers = new[] {
            new { a = 3, b = 6 },
            new { a = 4, b = 1 },
            new { a = 5, b = 8 }
        };
        foreach (var rec in integers)
            Console.Write(rec.a);
    }
}
```

**на экран будет выведено:**

7.

**В результате выполнения фрагмента программы:**

```
using System;

class Program {
    static void Main() {
        var integers = new[] {
            new { a = 3 },
            new { a = 4, b = 1 },
            new { a = 5, b = 8, c = 91 }
        };
        foreach (var rec in integers)
            Console.Write(rec.a);
    }
}
```

**на экран будет выведено:**

*Примечание:*

*Если возникнет ошибка компиляции, введите: \*\*\**

*Если ошибок и исключений нет, но на экран не выведется ничего, введите: ---*

*Если возникнет ошибка исполнения или исключение, введите: +++*

8.

**В результате выполнения фрагмента программы:**

```
using System;
using System.Collections.Generic;
using System.Linq;

class Program {
    public static void Main() {
        int[] numbers = { 2, 34, 92, 72, 09, 823, 71, 99, 31,
10 };

        IEnumerable<int> args =
            from n in numbers
            where n % 50 > 20 && (n ^ 10) < 30
            select n;

        foreach (var x in args)
            Console.Write(x);
        Console.Write(args.Count());
    }
}
```

**на экран будет выведено:**

*Примечание:*

*Если возникнет ошибка компиляции, введите: \*\*\**

*Если ошибок и исключений нет, но на экран не выведется ничего, введите: ---*

*Если возникнет ошибка исполнения или исключение, введите: +++*

9.

**Выберите все верные утверждения (укажите все верные ответы):**

- 1) IEnumerable – статический класс.
- 2) Интерфейс IEnumerable имеет как определённую реализацию, так и обобщённую реализацию.
- 3) При инициализации анонимной переменной обязательно ключевое слово **new**.
- 4) Все параметры анонимной переменной являются параметрами «только для чтения».
- 5) Анонимная переменная – объект ссылочного типа.

10.

**Верно, что объекты анонимного типа, могут быть объявлены в (укажите все верные ответы):**

- 1) Интерфейсе.
- 2) Методе.
- 3) Классе.
- 4) Структуре.
- 5) Конструкторе.

1	301
2	12
3	13
4	6879
5	22225
6	345
7	***
8	311
9	12345
10	2