1. **Вопрос:**

Может ли событие содержать ссылки на несколько методов?

**Ответ:** Да, может.

**Ссылки на источники:**

1) Обработка и создание событий: <https://docs.microsoft.com/ru-ru/dotnet/standard/events/>

1. **Вопрос:**

Выберите верные утверждения:

1) событие – тип

2) событие можно объявить в блоке исполняемого кода

3) событие всегда является членом класса, либо членом структуры

4) событие хранит ссылки на методы, как и многоадресный делегат

5) событие не может быть статическим

**Ответ:** 3, 4

**Ссылки на источники:**

1. События: <https://docs.microsoft.com/ru-ru/dotnet/csharp/programming-guide/events/index>
2. Ещё про события: <https://docs.microsoft.com/ru-ru/dotnet/csharp/language-reference/keywords/event>
3. **Вопрос:**

Модификаторами события могут быть(выберете верные варианты)

1) static

2) new

3) abstract

4) virtual

5) readonly

**Ответ:** 1, 2, 3, 4

**Ссылки на источники:**

1. Справочник по событиям: <https://docs.microsoft.com/ru-ru/dotnet/csharp/language-reference/keywords/event>
2. **Вопрос:**

Какие модификаторы видимости могут быть применены для событий?

**Ответ:** public, internal, protected, protected internal, private protected

**Ссылки на источники:**

1. Справочник по событиям: <https://docs.microsoft.com/ru-ru/dotnet/csharp/language-reference/keywords/event>
2. **Вопрос:**

Как подписаться на событие (методы, которые должны запуститься)

1. <КлассИлиОбъект>.<ИмяСобытия>()+=

<КлассЧейМетодДолженЗапуститься>.<МетодПодходящийПоСигнатуре>()

1. <ИмяСобытия>+=

<КлассЧейМетодДолженЗапуститься>.<МетодПодходящийПоСигнатуре>()

1. <ИмяСобытия>+=

<КлассЧейМетодДолженЗапуститься>.<МетодПодходящийПоСигнатуре>

1. <КлассИлиОбъект>.<ИмяСобытия>+=

<КлассЧейМетодДолженЗапуститься>.<МетодПодходящийПоСигнатуре>

1. <КлассИлиОбъект>.<МетодПодходящийПоСигнатуре>+=

<ИмяСобытия>

**Ответ:** 4

**Ссылки на источники:**

1. Подписка и отмена подписки на события: <https://docs.microsoft.com/ru-ru/dotnet/csharp/programming-guide/events/how-to-subscribe-to-and-unsubscribe-from-events>
2. **Вопрос:**

Что выведет программа?

delegate void MyEvent1(int n);

class MyEvent

{

public event MyEvent1 Event1;

public void OnEvent1(int n)

{

if (Event1 != null) Event1(n);

}

}

class Program

{

static void Main()

{

MyEvent ev =new MyEvent();

ev.Event1 += (n) => Console.Write(n++ \* 8);

ev.OnEvent1(3);

ev.OnEvent1(2);

}

}

**Ответ:** 2416

**Ссылки на источники:**

1. События: <https://docs.microsoft.com/ru-ru/dotnet/csharp/programming-guide/events/how-to-subscribe-to-and-unsubscribe-from-events>
2. **Вопрос:**

Встроенный делегат, обрабатывающий событие:

1)public delegate void EventHandler(object sender)

2)public delegate void EventHandler(EventArgs e)

3)public delegate void EventHandler(object sender, EventArgs e)

4)public delegate void EventHandler()

5)C# не имеет встроенных делегатов для этого

**Ответ:** 3

**Ссылки на источники:**

1. Про встроенные делегаты: <https://www.intertech.com/Blog/c-sharp-tutorial-understanding-c-events/>
2. **Вопрос:**

Выберите верные утверждения:

1)Для событий может использоваться модификатор области видимости internal

2)События могут реализовываться при помощи модификатора abstract

3)События НЕ могут быть членом класса static

4)События НЕ могут быть помечены модификатором sealed

**Ответ:** 1,2

**Ссылки на источники:**

1. Справочник по событиям: <https://docs.microsoft.com/ru-ru/dotnet/csharp/language-reference/keywords/event>
2. **Вопрос:**

Что будет выведено на экран после выполнения данного кода? (\*\*\* - ошибка компиляции) namespace ImSorry

{

class Program

{

delegate void del(int x);

private static event del growth = p\_1;

private static void p\_1(int y)

{

growth += (int z) => { Console.Write((int)z\*z);

};

}

static void Main(string[] args)

{

for (int i = 0; i <= 5; i++)

{

if (i % 2 == 1)

{

growth(i);

}

}

}

}

}

**Ответ:** 92525, т.к. мы в методе подписываемся на событие, при i = 1, мы 1 раз подпишемся, при i = 3, вызовется уже то, что подписали и добавится ещё одно, и тогда при i = 5, выведется 2 раза 25.

**Ссылки на источники:**

1. Подписка на события: <https://docs.microsoft.com/ru-ru/dotnet/csharp/programming-guide/events/how-to-subscribe-to-and-unsubscribe-from-events>
2. **Вопрос:**

- Что выведет следующая программа?

using System;

namespace Task-1

{

class MyEventArgs : EventArgs

{

public MyEventArgs() : base() { }

}

class Program

{

static event EventHandler OnEvent;

static void Main(string[] args)

{

Program pr = new Program();

OnEvent += MyMethod;

OnEvent(pr, EventArgs.Empty);

}

static void MyMethod(object sender, MyEventArgs args)

{ Console.WriteLine("This is my method!"); }

}

}

1) This is my method!

2) Пустую строку

3) Ничего - ошибка компиляции

**Ответ:** 3, т.к не выполняется контрвариантность

**Ссылки на источники:**

1. Контрвариантность: <https://metanit.com/sharp/tutorial/3.28.php>