



ZDOBĄDŹ WIEDZĘ Z ROZMÓW KWALIFIKACYJNYCH

20 ZDRADZIECKICH PYTAŃ Z C#

Poznaj najpopularniejsze pytania rekruterów .NET i C#,
które tylko z pozoru wydają się być oczywiste.

WSTĘP

Nazywam się Adam Kamizelich i od 2011 roku zajmuję się językiem C# oraz platformą .NET. Od 2014 roku tworzę rozproszone systemy IoT wspierane przez chmurę Microsoft Azure oraz mikroserwisy.

Na podstawie wielu rozmów w jakich miałem okazję uczestniczyć, po obydwu stronach stołu, postanowiłem zebrać listę pytań z C# oraz .NET, które moim zdaniem są zdecydowanie najciekawsze, a odpowiedzi tylko z pozoru wydają się być oczywiste dla kandydata.

Liczę, że podobnie jak ja polubisz język C#, a pytania z mojej listy już nigdy Cię nie zaskoczą no kolejnych rozmowach.

Adam Kamizelich



PYTANIE 1

Co będzie rezultatem wywołania poniższego fragmentu kodu?

```
string myString = (string)null;  
Console.WriteLine(myString is string);
```



PYTANIE 2

Co wyświetli się na ekranie konsoli po wywołaniu poniższego fragmentu kodu w funkcji Main?

```
Console.WriteLine(Math.Round(5.5));  
Console.WriteLine(Math.Round(6.5));
```



PYTANIE 3

Co wyświetli się na konsoli w wyniku działania programu?

```
static String str;  
static DateTime time;  
  
0 references  
static void Main(string[] args)  
{  
    Console.WriteLine(str == null ? "str == null" : str);  
    Console.WriteLine(time == null ? "time == null" : time.ToString());  
    Console.ReadLine();  
}
```



PYTANIE 4

Co użytkownik zobaczy na ekranie po uruchomieniu programu przedstawionego poniżej?

```
0 references
static Program()
{
    Console.WriteLine("static Program()");
}

0 references
public Program()
{
    Console.WriteLine("public Program()");
}

0 references
static void Main(string[] args)
{
    Console.WriteLine("static void Main(string[] args)");
    Console.ReadLine();
}
```



PYTANIE 5

Co wyświetli się na ekranie po uruchomieniu poniższego programu?

```
delegate void MyDelegate();

0 references
static void Main(string[] args)
{
    var delegates = new List<MyDelegate>();

    for (int i = 0; i < 6; i++)
    {
        delegates.Add(item: delegate { Console.WriteLine(i); });
    }

    foreach (var @delegate in delegates)
    {
        @delegate();
    }
}
```



PYTANIE 6

Co wyświetli się na ekranie konsoli?

```
[Flags]
1 reference
public enum Status
{
    Funny = 0x01,
    Hilarious = 0x02,
    Boring = 0x04,
    Cool = 0x08,
    Interesting = 0x10,
    Informative = 0x20,
    Error = 0x40
}

0 references
public static void Main(string[] args)
{
    var code = 24;
    Console.WriteLine(
        String.Format("This Quiz is: {0}", (Status)code));
}
```



PYTANIE 7

Co się stanie po uruchomieniu danego fragmentu kodu?

```
0 references
static void Main(string[] args)
{
    float q = float.MaxValue;
    float w = float.MaxValue;

    checked
    {
        float a = q * w;
        Console.WriteLine(a.ToString());
    }
}
```



PYTANIE 8

Co wyświetli się na ekranie po uruchomieniu poniższego programu?

```
1 reference
class ClassA
{
    0 references
    public void Run(int q)
    {
        Console.WriteLine("Run from ClassA");
    }
}

2 references
class ClassB : ClassA
{
    1 reference
    public void Run(double p)
    {
        Console.WriteLine("Run from ClassB");
    }
}

0 references
static void Main(string[] args)
{
    int i = 5;
    ClassB classB = new ClassB();
    classB.Run(i);
    Console.ReadLine();
}
```



PYTANIE 9

Co wyświetli się na ekranie po uruchomieniu przez użytkownika poniższego programu?

```
2 references
class X { }

2 references
class Y : X { }

1 reference
class ClassA
{
    0 references
    public void Run(Y y)
    {
        Console.WriteLine("Run from ClassA");
    }
}

2 references
class ClassB : ClassA
{
    1 reference
    public void Run(X x)
    {
        Console.WriteLine("Run from ClassB");
    }
}

0 references
static void Main(string[] args)
{
    ClassB classB = new ClassB();
    classB.Run(x: new Y());
    Console.ReadLine();
}
```



PYTANIE 10

Co wyświetli się na ekranie po uruchomieniu poniższego programu?

```
0 references
class Program
{
    0 references
    static void Main(string[] args)
    {
        MyClassB b = new MyClassB();
        MyClassA a = b;
        a.Run();
        Console.ReadLine();
    }
}

3 references
class MyClassA
{
    0 references
    public MyClassA()
    {
        Console.WriteLine("constructor A");
    }

    1 reference
    public void Run()
    {
        Console.WriteLine("A");
    }
}

3 references
class MyClassB : MyClassA
{
    1 reference
    public MyClassB()
    {
        Console.WriteLine("constructor B");
    }

    0 references
    public void abc()
    {
        Console.WriteLine("B");
    }
}
```




PYTANIE 11

Jaki będzie rezultat po uruchomieniu programu?

```
0 references
class Program
{
    1 reference
    static bool A()
    {
        Console.WriteLine("A");
        return false;
    }

    1 reference
    static bool B()
    {
        Console.WriteLine("B");
        return true;
    }

    0 references
    static void Main(string[] args)
    {
        if (A() & B())
        {
            Console.WriteLine("I'm in if");
        }
    }
}
```



PYTANIE 12

Co wyświetli się na ekranie po uruchomieniu funkcji Main?

```
0 references
static void Main(string[] args)
{
    int x = 5;
    decimal y = x / 12;
    Console.WriteLine(y.ToString());
}
```



PYTANIE 13

Czy po usunięciu metody

"public static Point operator +(Point p1, Point p2)"

kod nadal będzie się kompilował?

Dlaczego tak/nie?

```
0 references
class Program
{
    0 references
    static void Main(string[] args)
    {
        Point ptOne = new Point(xPos: 15, yPos: 20);
        Point ptTwo = new Point(xPos: 40, yPos: 50);
        Point ptThree = ptOne + ptTwo;
    }
}

10 references
public class Point
{
    3 references
    public int X { get; set; }
    3 references
    public int Y { get; set; }
    3 references
    public Point(int xPos, int yPos)
    {
        X = xPos;
        Y = yPos;
    }
    1 reference
    public static Point operator +(Point p1, Point p2)
    {
        return new Point(xPos: p1.X + p2.X, yPos: p1.Y + p2.Y);
    }
}
```



PYTANIE 14

Jakie parametry zostaną przekazane w metodzie Foo

```
0 references
class Program
{
    0 references
    static void Main(string[] args)
    {
        Foo(y:17, x: 1);
    }

    1 reference
    static int Foo(int x, int y)
    {
        return (x - y) * (x + y);
    }
}
```



PYTANIE 15

Jaki będzie rezultat działania programu?

```
0 references
static void Main(string[] args)
{
    goto temp;
    int i = 5;
    { Console.WriteLine(i); }
    temp:
    Console.WriteLine("Good bye!");
    Console.ReadLine();
}
```



PYTANIE 16

Co wyświetli się na ekranie po uruchomieniu poniższego programu?

```
0 references
class Program
{
    0 references
    static void Main(string[] args)
    {
        int someInt;
        B(out someInt);
        Console.WriteLine(someInt);

        A(ref someInt);
        Console.WriteLine(someInt);

        C(someInt);
        Console.WriteLine(someInt);

        Console.ReadLine();
    }

    1 reference
    static void A(ref int value)
    {
        value = 1;
    }

    1 reference
    static void B(out int value)
    {
        value = 2;
    }

    1 reference
    static void C(int value)
    {
        value = 0;
    }
}
```



PYTANIE 17

Co wyświetli się na ekranie po uruchomieniu programu zaprezentowanego poniżej?

```
0 references
static void Main(string[] args)
{
    Queue queue = new Queue();
    queue.Enqueue(obj: 1);
    queue.Enqueue(obj: 2);
    queue.Enqueue(obj: 3);

    foreach (var obj:object? in queue)
    {
        Console.WriteLine(obj);
    }

    Console.WriteLine("How many elements in the Queue? -> " + queue.Count);
    Console.WriteLine("Does the Queue contain 3? -> " + queue.Contains(obj: 3));
}
```



PYTANIE 18

Co wyświetli się na ekranie po uruchomieniu poniższego programu?

```
0 references
public static void Main(string[] args)
{
    byte num = 100;
    dynamic val = num;
    Console.WriteLine(val.GetType());
    val += 100;
    Console.WriteLine(val.GetType());
}
```



PYTANIE 19

Co wyświetli się na konsoli po uruchomieniu poniższego programu?

```
0 references
class Program
{
    0 references
    public static void Main(string[] args)
    {
#if (!pi)
        Console.WriteLine("i");
#else
        Console.WriteLine("PI undefined");
#endif

        Console.WriteLine("ok");
        Console.ReadLine();
    }
}
```



PYTANIE 20

Co będzie rezultatem działania programu?

```
0 references
public static void Main(string[] args)
{
    int[] arr = new int[2];
    arr[1] = 10;
    Object o = arr;
    int[] arr1 = (int[])o;
    arr1[1] = 100;
    Console.WriteLine(arr[1]);
    ((int[])o)[1] = 1000;
    Console.WriteLine(arr[1]);
}
```



ODPOWIEDZI

Nie znasz odpowiedzi, na któreś z pytań i potrzebujesz pomocy? Napisz do mnie, a wyślę Ci wszystkie odpowiedzi na Twoją skrzynkę email.

adam@gotoit.pl

Zapraszam do kontaktu!
Adam Kamizelich