Choose the namespace in which the interface IEnumerable is declared?

A. System.Collections

C. Both A and B

D. None of the mentioned

B. System.Collections.Generic

For the given set of code what is output?

```
class Program
{
    static void Main(string[] args)
    {
        int[] nums = { 1, -2, 3, 0, -4, 5 };
        var posNums = nums.Where(n => n < 10).Select(r => r%3);
        Console.Write("The values in nums: ");
        foreach (int i in posNums) Console.Write(i + " ");
        Console.WriteLine();
        Console.ReadLine();
    }
}
```

Run time error

Compile time error

2 -1 0 0 -2 1

1 -2 0 0 -1 2

Choose the

System.Text

System.Collections.Generic

System.Linq.Expressions

System.Linq

For the given set of code what is output?

```
class Program
  {
    static void Main(string[] args)
    {

      int[] nums = { 1, -2, -3, 5 };
      var posNums = from n in nums
                    orderby n descending
                    select n*4 / 2;
      Console.Write("The values in nums: ");
      foreach (int i in posNums) Console.Write(i + " ");
      Console.WriteLine();
      Console.ReadLine();
    }
  }
```

Run time error

10 2 -4 -6

1 5 -2 -3

5 1 -2 -3

For the given set of code what is output representing?

```
class Program
{
   static void Main(string[] args)
   {
      int[] nums = { 1, -2, 3, 0, -4, 5 };
      var posNums = from n in nums
                    where n > 0
                    select n;
      int len = posNums.Count();
      Console.WriteLine(len);
      Console.ReadLine();
   }
}
```

Execution of code with nothing being printed

Run time error

Execution of code with counting total numbers greater than zero

Execution of code with printing all numbers

Select the output for given code snippet:

```
class Program
{
```

```
static void Main(string[] args)
{
int[] nums = {1};
var posNums = from n in nums
where n > 0
select Math.Max(78, 9);
Console.Write("The largest values in nums: ");
foreach (int i in posNums) Console.Write(i + " ");
Console.WriteLine();
Console.ReadLine();
}
}
```

code run successfully print nothing

Compile time error

code run successfully print required output

Run time error

What will be the output of given code snippet?

```
class Program
{
    static void Main(string[] args)
    {
        int[] nums = { 1, -2, 3, 0, -4, 5};
        var posNums = from n in nums
```

```
                where n >= 0
                select n;
        foreach (int i in posNums)
        Console.Write(i + " ");
        Console.WriteLine();
        Console.ReadLine();
    }
}
```

1, 3, 5

Run time error

0, 1, -2, -4, 5

1, 3, 0, 5

Select the output for given code snippet:

```
class Program
{
 static void Main(string[] args)
 {
 int[] nums = { 16, 9, 25};
 var posNums = from n in nums
 where n > 0
 select Math.Sqrt(n);

 Console.Write("The Square root values in nums: ");
```

```
 foreach (int i in posNums) Console.Write(i + " ");
 Console.WriteLine();
 Console.ReadLine();
 }
}
```

Run time error

code run successfully print nothing

code run successfully print required output

Compile time error

Select the output for given code snippet:

```
class Program
{
 static void Main(string[] args)
 {
 int[] nums = { 1, -2, 3, 0, -4, 5 };
 var posNums = from n in nums
 where n > -5 && n < 6
 orderby n descending
 select n;
 Console.Write("Descending order in nums: ");
 foreach (int i in posNums) Console.Write(i + " ");
 Console.WriteLine();
 Console.ReadLine();
```

}

}

Compile time error

Print nothing code run successfully

Arranged in descending order code run successfully

Run time error

Please read the questions carefully and choose the most appropriate option. Which of the given options are TRUE about the String Class in C#.NET?

A string built using String Class is Immutable.

All the listed options

A string built using StringBuilder Class is Mutable.

Two strings can be concatenated by using an expression of the form s3 = s1 + s2;

Please read the questions carefully and choose the most appropriate option.Which of the following components of the .NET framework provide an extensible set of classes that can be used by any .NET compliant programming language?

1..NET class libraries

2.Component Object Model

only 2

Both 1 and 2

only 1

None of the listed options

The output of code is ?

```
class test
{
public void print()
{
Console.WriteLine("Csharp:");
}
}
class Program
{
static void Main(string[] args)
{
test t;
t.print();
Console.ReadLine();
}
}
```

Code run and print “Csharp”

None of the mentioned

Code run successfully print nothing

Syntax error as t is unassigned variable which is never used

Please read the questions carefully and choose the most appropriate option.Which of the given options are TRUE?

Member function of a class are by default private.

All the listed options

A private function of a class can access a public function within the same class.

Data members of a class are by default private.

Please read the questions carefully and choose the most appropriate option.Static procedures can access instance data? State TRUE or FALSE?

false

true

Select the output for following set of code :

```
class sample
{
    public int i;
    public int j;
    public void fun(int i, int j)
    {
        this.i = i;
        this.j = j;
    }
}
class Program
{
    static void Main(string[] args)
    {
        sample s = new sample();
```

```
        s.i = 1;
        s.j = 2;
        s.fun(s.i, s.j);
        Console.WriteLine(s.i + " " + s.j);
        Console.ReadLine();
    }
}
```

Run successfully but prints nothing

1 2

Error as ‘this’ reference would not be able to call ‘i’ and ‘j’

Error while calling s.fun() due to inaccessible level

Select output for following set of code.

```
class sample
{
    public int i;
    public int[] arr = new int[10];
    public void fun(int i, int val)
    {
        arr[i] = val;
    }
}
class Program
{
    static void Main(string[] args)
```

```
    {
        sample s = new sample();
        s.i = 10;
        sample.fun(1, 5);
        s.fun(1, 5);
        Console.ReadLine();
    }
}
```

sample.fun(1, 5) will set value as 5 in arr[1]

sample.fun(1, 5) will not work correctly

s.i = 10 cannot work as i is ‘public’

s.fun(1, 5) will work correctly

What do the following code implies ?

csharp abc;

abc = new csharp();

create an object of type csharp on stack

Create an object of type csharp on heap or on stack depending on whether csharp is class or struct

Object creation on class csharp

Create an object of type csharp on heap or on stack depending on size of object

Please read the questions carefully and choose the most appropriate option.Which of the given options CANNOT be a target for custom attribute?

Event

Namespace

Delegate

All the listed options

Select output for following set of code.

```
class sample
{
    public int i;
    public int[] arr = new int[10];
    public void fun(int i, int val)
    {
        arr[i] = val;
    }
}
class Program
{
    static void Main(string[] args)
    {
        sample s = new sample();
        s.i = 10;
        sample.fun(1, 5);
        s.fun(1, 5);
        Console.ReadLine();
    }
}
```

sample.fun(1, 5) will not work correctly

s.fun(1, 5) will work correctly

sample.fun(1, 5) will set value as 5 in arr[1]

s.i = 10 cannot work as i is ‘public’

Select output for following set of code :

CONTROL STRUCTURES

```
static void Main(string[] args)
{
   int x;
   for (x = 10; x <= 15; x++)
   while (Convert.ToBoolean(Convert.ToInt32(x)))
   {
      do
      {
         Console.WriteLine(1);
         if (Convert.ToBoolean(x >> 1))
         continue;
      }while (Convert.ToBoolean(0));
      break;
   }
   Console.ReadLine();
}
```

0 0 0….infinite times

System outofflow exception error.

1 1 1 1 1 1 :ans

1 1 1….infinite times

Select output for following set of code :

```
static void Main(string[] args)
{
    int i, s = 0, a = 1, d;
    i = Convert.ToInt32(Console.ReadLine());
    do
    {
        d = i % (2 * 4);
        s = s + d * a;
    }while ((Convert.ToInt32(i = i / (2 * 4))) != 0 && (Convert.ToBoolean(Convert.ToInt32((a) = (a * 10)))));
    Console.WriteLine(s);
    Console.ReadLine();
}
```

enter i = 342.

It finds reverse of i

It finds binary equivalent of i

It finds sum of digits of i

It finds octal equivalent of i

Which of these is not a correct statement?

Recursion always uses stack

Recursive methods are faster that programmers written loop to call the function repeatedly using a stack

Recursion is always managed by C# Runtime environment

A recursive method must have a base case

What will be the output for given set of code?

```
static void Main(string[] args)
      {
         int n = 1;
         method(n);
         Console.WriteLine(n);
         method1(ref n);
         Console.WriteLine(n);
         Console.ReadLine();
      }
      static void method(int num)
      {
         num += 20;
         Console.WriteLine(num);
      }
      static void method1(ref int num)
      {
         num += 20;
         Console.WriteLine(num);
      }
```

1

1

1

1

11

21

21

11

21

1

21

21

21

21

21

21

What will be the output for given set of code?

```
static void Main(string[] args)
{
int []arr = new int[]{ 1, 2, 3, 4, 5};
fun (ref arr);
for (int i = 0; i < arr.Length ; i++)
Console.WriteLine( arr[i] + " ");
}
static void fun(ref int[]a)
{
a = new int[6];
a[3] = 32;
a[1] = 24;
}
```

0

24

0

32

0

0

0

0

32

0

0

0

24

0

32

0

0

0

0

0

32

0

0

0

What is Recursion in CSharp defined as?

Recursion another process of defining a method that calls other methods repeatedly

Recursion another form of class

Recursion is a process of defining a method that calls other methods which in turn call again this method

Recursion is a process of defining a method that calls itself repeatedly

Please read the questions carefully and choose the most appropriate option.Which of the given options are TRUE?

1.A switch statement can act on numerical as well as Boolean types.

2.A switch statement can act on characters, strings and enumerations types.

None of the listed options

only 1

Both 1 and 2

only 2

What will be the correct output for given code snippet?

```
class maths
 {
 public int fact(int n)
 {
 int result;
 if (n == 1)
 return 1;
 result = fact(n - 1) * n;
```

```
return result;
}
}
class Output
{
static void Main(String[] args)
{
maths obj = new maths() ;
Console.WriteLine(obj.fact(4)*obj.fact(2));
}
}
```

60

48

64

120

Which method does following set of code explains?

```
static void Main(string[] args)
{
    int a = 10, b = 20;
    method(ref a,  ref b);
    console.writeline(a + "  " + b);
}
static void swap(ref int i,  ref int j)
{
    int t;
```

```
    t = i;
    i = j;
    j = t;
}
```

Call by reference

Call by value

parameter arrays

Output parameter

What will be the correct output for given code snippet?

```
class maths
  {
    public int fact(int n)
    {
      int result;
      if (n == 1)
        return 1;
      result = fact(n - 1) * n;
      return result;
    }
  }
  class Output
  {
    static void Main(String[] args)
    {
      maths obj = new maths();
```

```
            Console.WriteLine(obj.fact(1));
            Console.ReadLine();
        }
    }
```

1

10

2

0

What is output of following set of code?

```
static void Main(string[] args)
{
 Program p = new Program();
 p.display(2, 3, 8);
 int []a = { 2, 56, 78, 66 };
 Console.WriteLine("example of array");
 Console.WriteLine("elements added are");
 p.display(a);
 Console.ReadLine();
}
public void display(params int[] b)
{
 foreach (int i in b)
 {
 Console.WriteLine("ARRAY IS HAVING:{0}", i);
 }
```

}

Code runs successfully but prints nothing

Code runs successfully and prints given on console

Run time error

Compile time error

Please read the questions carefully and choose the most appropriate option.An enum that is declared inside a class, struct, namespace or interface is treated as public. State True or False.

false

true

What will be the output of given code snippet?

```
class Program
{
static void Main(string[] args)
{
char []chars = {'a', 'b', 'c'};
String s = new String(chars);
Console.WriteLine(s);
Console.ReadLine();
}
}
```

c

a

abc

b

What will be the output of give code snippet?

```
class Program
{
 static void Main(string[] args)
 {
 String s1 = "Hello i love Csharp";
 StringBuilder s2 = new StringBuilder(s1);
 Console.WriteLine(s1.Equals(s2));
 Console.ReadLine();
 }
}
```

Compile time error

true

False

0

Choose the correct output for given set of code?

```
  enum per
 {
     a,
```

```
    b,
    c,
    d,
}
per.a = 10;
Console.writeline(per.b);
```

2

11

Compile time error

1

Select correct declaration of defining array of parameters:

```
void func(int[] x)
{
}

void func(int x)
{
}
void fun(param int[] x)
{
}

void func(param int[])
{
}
```

Please read the questions carefully and choose the most appropriate option.Which of the given options are TRUE about enumerators?

1.The value of each successive enumerator is decreased by 1.

2.Values of enum elements cannot be populated from a database.

None of the listed options

only 1

only 2

Both 1 and 2

What will be the output of given code snippet?

```
static void Main(string[] args)
{
 string s1 = " Ixg";
 string s2 = s1.Insert(3,"i");
 string s3 = s2.Insert(5, "o");
 for (int i = 0; i < s3.Length; i++)
 Console.WriteLine(s3[i]);
 Console.ReadLine();
}
```

Ixgo

Ixig

Ixigo

Ixigo

Please read the questions carefully and choose the most appropriate option.What is the size of a Decimal data type?

32 byte

4 byte

16 byte

8 byte

What will be the output of given set of code?

```
static void Main(string[] args)
{
    int[] x = {65, 66, 67, 68, 69, 70};
    fun(x);
    Console.ReadLine();
}
static void fun(params int[] b )
{
    int i;
    for (i = 5; i > 0 ; i--)
    {
        b[i] = b[i] + 32;
        Console.WriteLine(Convert.ToChar(b[i]));
    }
}
```

F, E, D, C, B, A

A, B, C, D, E, F

f, e, d, c, b

b, c, d, e, f

Access specifiers

What will be size of object created depicted by csharp code snippet?

```
class baseclass
{
private int a;
protected int b;
public int c;
}
class derived : baseclass
{
private int x;
protected int y;
public int z;
}
class Program
{
static Void Main(string[] args)
{
derived a = new derived();
}
}
```

- 20 bytes
- 24 bytes
- 16 bytes
- 12 bytes

What will be the output of given code snippet?

```
class access
{
    public int x;
    private int y;
    public  void cal(int a, int b)
    {
        x = a + 1;
        y = b;
    }
}
class Program
{
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
    static void Main(string[] args)
    {
        access obj = new access();
        obj.cal(2, 3);
        Console.WriteLine(obj.x + " " + obj.y);
    }
}
```

- Run time error
- Compile time error
- 3 3
- 2 3

What will be the output of given code snippet?

```
class access
{
    public int x;
    private int y;
    public  void cal(int a, int b)
    {
        x = a + 1;
        y = b;
    }
    public  void print()
    {
        Console.WriteLine(" " + y);
    }
}
class Program
{
    static void Main(string[] args)
    {
        access obj = new access();
        obj.cal(2, 3);
        Console.WriteLine(obj.x);
        obj.print();
        Console.ReadLine();
    }
}
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

- Run time error
- 3 3
- 2 3
- Compile time error

What will be the output of following set of code?

```
class sum
{
    public int x;
    public int y;
    public  int add (int a, int b)
    {
        x = a + b;
        y = x + b;
        return 0;
    }
}
class Program
{
    static void Main(string[] args)
    {
        sum obj1 = new sum();
        sum obj2 = new sum();
        int a = 2;
        obj1.add(a, a + 1);
        obj2.add(5, a);
        Console.WriteLine(obj1.x + "  " + obj2.y);
        Console.ReadLine();
    }
}
```

- 9, 10
- 3, 2
- 5, 9
- 6, 9

What will be the output of following set of code?

```
class static_out
```

```
{
   public static int x;
   public  static int y;
   public int add(int a, int b)
   {
      x = a + b;
      y = x + b;
      return 0;
   }
}
class Program
{
   static void Main(string[] args)
   {
      static_out obj1 = new static_out();
      static_out obj2 = new static_out();
      int a = 2;
      obj1.add(a, a + 1);
      obj2.add(5, a);
      Console.WriteLine(static_out.x + " " + static_out.y );
      Console.ReadLine();
   }
}
```

- 6 6
- 7 7
- 7 9
- 9 7

What will be the output of following set of code?

```
class sum
{
   public int x;
   private int y;
   public void math(int a, int b)
   {
      x = a * 4;
      y = b;
   }
}
class Program
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
{
   static void Main(string[] args)
   {
      sum p = new sum();
      p.math(12, 30);
      Console.WriteLine(p.x + "  " + p.y);
      Console.ReadLine();
   }
}
```

- 0, 0
- 48, 0
- 48, 30
- Compile time error

Accessibility modifier defined in a class are?

- public, internal, protected internal.
- public, private, internal, protected internal.
- public, private, protected
- public, private, protected, internal, protected internal

What will be the output of code?

```
class math
{
   public int a,b;
   public math(int i,  int j)
   {
      a = i;
      b = j;
   }
   public  void sum(math m)
   {
      m.a *= 2;
      m.b += 2;
   }
}
class Program
```

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
{
   static void Main(string[] args)
   {
      math t = new math(20,  10);
      t.sum(t);
      Console.WriteLine(t.a + "  " + t.b);
      Console.ReadLine();
   }
}
```

- 40, 12
- 5, 40
- 10, 20
- 20, 10

```
class test
{
   public   int a;
   public  int b;
   public  test(int i, int j)
   {
      a = i;
      b = j;
   }
   public void meth(test o)
   {
      o.a *= 2;
      o.b /= 2;
   }
}
class Program
{
   static void Main(string[] args)
   {
      test obj = new test(10, 20);
      obj.meth(obj);
      Console.WriteLine(obj.a + " " + obj.b);
      Console.ReadLine();
   }
}
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

- 40, 20
- 10, 20
- 20, 40
- 20, 10

Which of these is used as default specifier for a member of class if no access specifier is used for it?

- protected
- public, within its own class
- private
- public

Basics

Choose the class on which all stream classes are defined?

- All of the mentioned
- Sytem.Input.stream
- System.Output.stream
- System.IO.stream

A method used to write a single byte to an output stream?

- Read()
- write()
- void WriteByte(byte value)
- int Write(byte[] buffer ,int offset ,int count)

Please read the questions carefully and choose the most appropriate option.Which of the given options a

1.The attributes applied can be read from an assembly using Reflection class.
2.An attribute can have parameters.

Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...

○ only 1

○ None of the listed options

● Both 1 and 2

○ only 2

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

Please read the questions carefully and choose the most appropriate option.Which of the given options a
1.Arrays can be rectangular or jagged.
2.Rectangular arrays have similar rows stored in adjacent memory locations.

○ None of the listed options

○ only 2

○ only 1

● Both 1 and 2

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

Please read the questions carefully and choose the most appropriate option.Read the below statements ca

Statement 1: Managed code is the code that is compiled by the JIT compilers.
Statement 2: Managed code is the code where resources are Garbage Collected.

Which of the above statements are TRUE about Managed Code?

○ All statements are true

○ Only Statement 2 is true

● None of the statements is true

○ Only Statement 1 is true

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

Please read the questions carefully and choose the most appropriate option.Read the below statements c

Statement 1: CLR provides services to run "managed" applications.
Statement 2: The resources are garbage collected.
Statement 3: CLR provides services to run "unmanaged" applications.

Which of the following statements are TRUE about the .NET CLR?

○ All statements are true

● Only Statements 1 and 2 are true

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

- Only Statement 3 is true
- Only Statement 2 is true

Choose the filemode method which is used to create a new output file with condition that file with same

- FileMode.OpenOrCreate
- FileMode.Truncate
- FileMode.Create
- FileMode.CreateNew

Please read the questions carefully and choose the most appropriate option.Which of the given options a

1.By position
2.By name

- only 1
- only 2
- Both 1 and 2
- None of the listed options

Which among is used for storage of memory aspects?

- BufferedStream
- None of the mentioned
- MemoryStream
- FileStream

Class

Please read the questions carefully and choose the most appropriate option.Which of the given options a

- Static functions of a class never receive the 'this' reference.

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

- this' reference can be modified in the instance member function of a class.
- All the listed options
- this' reference continues to exist even after control returns from an instance member function.

Please read the questions carefully and choose the most appropriate option.With which of the given op
1.Static data
2.Instance data

- only 1
- only 2
- Both 1 and 2
- None of the listed options

Please read the questions carefully and choose the most appropriate option.Which of the given options

- All the listed options
- Data members of a class are by default private.
- A private function of a class can access a public function within the same class.
- Member function of a class are by default private.

Select the output for following set of code :

```
class sample
{
   public int i;
   public int j;
   public void fun(int i, int j)
   {
      this.i = i;
      this.j = j;
   }
}
class Program
{
   static void Main(string[] args)
   {
      sample s = new sample();
      s.i = 1;
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt

```
        s.j = 2;
        s.fun(s.i, s.j);
        Console.WriteLine(s.i + " " + s.j);
        Console.ReadLine();
    }
}
```

- Error as ‘this’ reference would not be able to call ‘i’ and ‘j’
- Run successfully but prints nothing
- Error while calling s.fun() due to inaccessible level
- 1 2

Select the output for following set of code :

```
class z
{
public string name1;
public string address;
public void show()
{
Console.WriteLine("{0} is in {1}", name1, address);
}
}
class Program
{
static void Main(string[] args)
{
z n = new z();
n.name1 = "harsh";
n.address = "new delhi";
n.show();
Console.ReadLine();
}
}
```

- Run successfully prints nothing
- Syntax error
- {0} is in city{1} harsh new delhi
- harsh is in new delhi

What is most specified using class declaration ?

- scope

Formatted: Font: Times New Roman
Formatted Table
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

- type & scope
- type
- None of mentioned

Please read the questions carefully and choose the most appropriate option
.Which of the given options CANNOT be a target for custom attribute?

- All the listed options
- Delegate
- Namespace
- Event

Please read the questions carefully and choose the most appropriate option.The string built using the Strin
class are mutable. State TRUE or FALSE.

- true
- false

Select the output for following set of code :

```
class sample
{
   public int i;
   public int j;
   public void fun(int i, int j)
   {
      this.i = i;
      this.j = j;
   }
}
class Program
{
   static void Main(string[] args)
   {
      sample s = new sample();
      s.i = 1;
      s.j = 2;
      s.fun(s.i, s.j);
      Console.WriteLine(s.i + " " + s.j);
      Console.ReadLine();
   }
}
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt

- Error while calling s.fun() due to inaccessible level
- Run successfully but prints nothing
- Error as ‘this’ reference would not be able to call ‘i’ and ‘j’
- 1 2

Select output for following set of code.

```
class sample
{
   public int i;
   public int[] arr = new int[10];
   public void fun(int i, int val)
   {
      arr[i] = val;
   }
}
class Program
{
   static void Main(string[] args)
   {
      sample s = new sample();
      s.i = 10;
      sample.fun(1, 5);
      s.fun(1, 5);
      Console.ReadLine();
   }
}
```

- sample.fun(1, 5) will set value as 5 in arr[1]
- s.fun(1, 5) will work correctly
- sample.fun(1, 5) will not work correctly
- s.i = 10 cannot work as i is ‘public’

The output of code is ?

```
class test
{
public void print()
{
Console.WriteLine("Csharp:");
}
}
class Program
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt

```
{
static void Main(string[] args)
{
test t;
t.print();
Console.ReadLine();
}
}
```

- None of the mentioned
- Code run and print “Csharp”
- Code run successfully print nothing
- Syntax error as t is unassigned variable which is never used

Select the output for following set of code :

```
class z
{
public int X;
public int Y;
public const int c1 = 5;
public const int c2 = c1 * 25;
public void set(int a, int b)
{
X = a;
Y = b;
}

}
class Program
{
static void Main(string[] args)
{
z s = new z();
s.set(10, 20);
Console.WriteLine(s.X + " " + s.Y);
Console.WriteLine(z.c1 + " " + z.c2);
Console.ReadLine();
}
}
```

- 20 10
- 20 10
- 10 20

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

- [x] 10 20
- [ ] 5 125

What do the following code implies ?

```
csharp abc;
abc = new csharp();
```

- [ ] Object creation on class csharp
- [ ] Create an object of type csharp on heap or on stack depending on size of object
- [ ] create an object of type csharp on stack
- [x] Create an object of type csharp on heap or on stack depending on whether csharp is class or struct

What is most specified using class declaration ?

- [ ] None of mentioned
- [ ] type & scope
- [ ] type
- [ ] scope

Please read the questions carefully and choose the most appropriate option.Static procedures can access i

- [x] true
- [ ] false

Please read the questions carefully and choose the most appropriate option.Which of the following compo si
any .NET compliant programming language?

1..NET class libraries

2.Component Object Model

- [ ] only 2
- [x] Both 1 and 2
- [ ] None of the listed options
- [ ] only 1

Console

What is output returned by Console if ReadLine() stores I/O error?

Formatted ... (repeated 50 times)

- false
- 0
- 1
- I/O EXCEPTION ERROR

Choose the output returned when error condition generates while read() reads from the console.

- 0
- -1
- false
- All the given options

Which among the following methods used writes characters to a string?

- StringWriter
- StreamReader
- StreamWriter
- None

Select the correct input methods provided by Console?

- ReadLine()
- ReadKey(), ReadLine()
- Read(), ReadLine()
- Read(), ReadLine(), ReadKey()

Which method in Console enables to read individual inputs directly from the keyboard in a non line buf

- ReadKey()
- ReadLine()
- All the given options

Formatted
Formatted Table
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted Table
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted Table
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted Table
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted Table
Formatted
Formatted
Formatted
Formatted
Formatted

- Read()

Which of these method used to read string from the console?

- readLine()
- read()
- get()
- getline()

Which of these method/methods used to read block or array of bytes from the file?

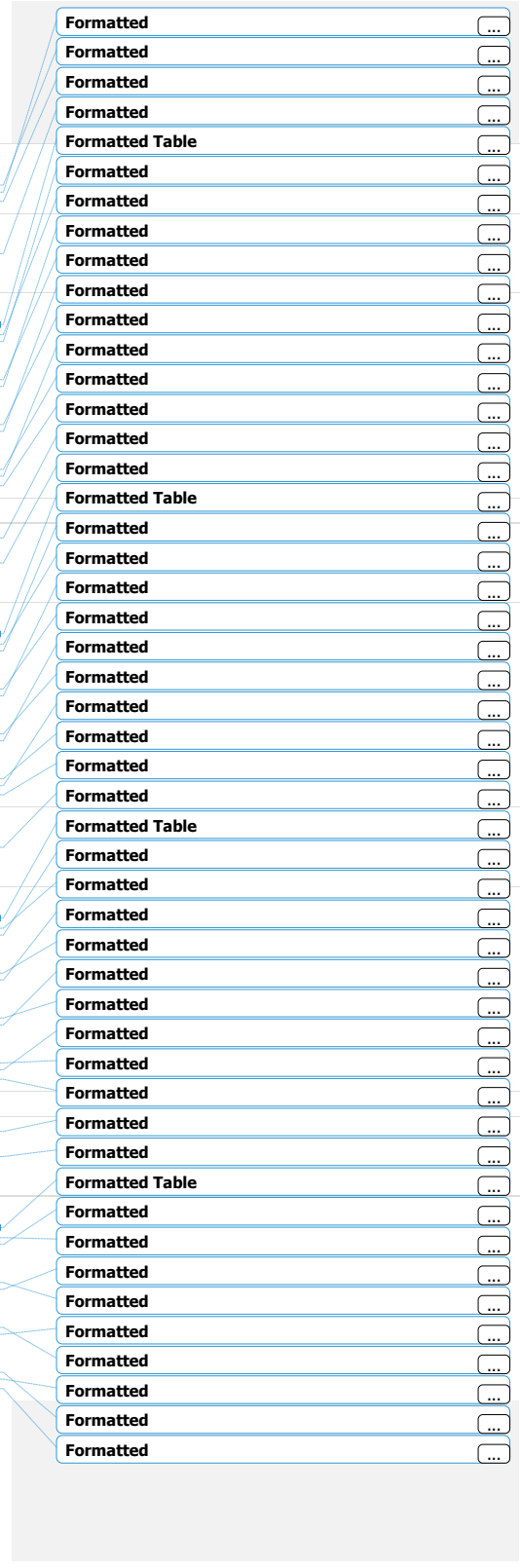
- ReadByte()
- Read()
- Readkey()
- ReadLine()

Choose the object of TextReader class.

- Console.In
- Console.Error
- None
- Console.Out

Choose the methods provided by Console.Out and Console.Error?

- ReadKey
- WriteKey
- ReadLine
- WriteLine



Choose the output return when read() reads the character from the console?

- Boolean
- String
- Char
- Integer

Choose the output for following set of code?

```
 static void Main(string[] args)
 {
Console.WriteLine("This is a Console Application:");
Console.Write("Please enter your lucky number:");
    string val1 = Console.ReadLine();
int val2 = System.Convert.ToInt32(val1, 10);
    val2 = val2 * val2;
Console.WriteLine("square of number is:" +val2);
Console.Read();
 }
```

- Run successfully ask for input and hence display the results
- Compile time error
- Run successfully donot prints anything
- Syntax Error

What would be the output for following input from the console as a character?

```
 static void Main(string[] args)
 {
Console.WriteLine("what is your name?");
    char s;
    s = Convert.ToChar(Console.ReadLine());
Console.WriteLine("how are you: "+s);
Console.Read();
 }
```

- Run time error
- Code run successfully prints nothing on console
- Compile time error
- Code run successfully prints input on console

Name the exception thrown by read() on failure.

- SystemInputException
- InterruptedException
- I/O Exception
- SystemException

Which of these methods are used to read single character from the console?

a) get()
b) getline()
c) read()
d) readLine()

Console

What is output returned by Console if ReadLine() stores I/O error?

- false
- 0
- 1
- I/O EXCEPTION ERROR

Choose the output returned when error condition generates while read() reads from the console.

- 0
- -1

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

- false
- All the given options

Which among the following methods used writes characters to a string?

- StringWriter
- StreamReader
- StreamWriter
- None

Select the correct input methods provided by Console?

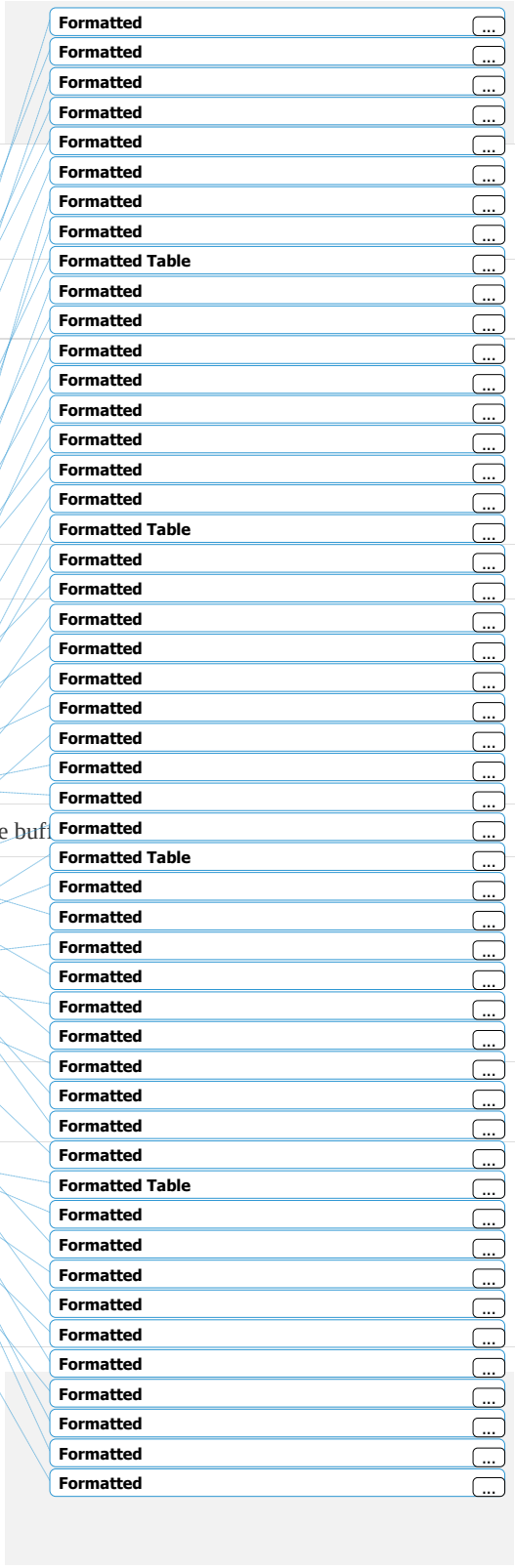
- ReadLine()
- ReadKey(), ReadLine()
- Read(), ReadLine()
- Read(), ReadLine(), ReadKey()

Which method in Console enables to read individual inputs directly from the keyboard in a non line buf

- ReadKey()
- ReadLine()
- All the given options
- Read()

Which of these method used to read string from the console?

- readLine()
- read()
- get()
- getline()



Which of these method/methods used to read block or array of bytes from the file?

- ReadByte()
- Read()
- Readkey()
- ReadLine()

Choose the object of TextReader class.

- Console.In
- Console.Error
- None
- Console.Out

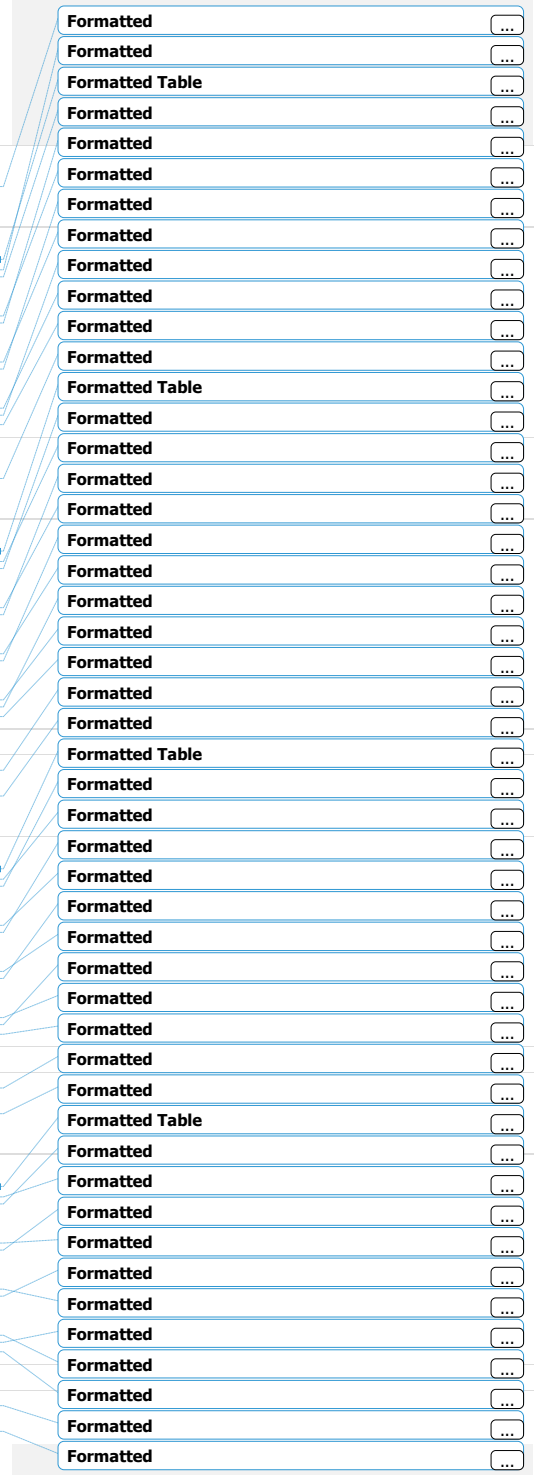
Choose the methods provided by Console.Out and Console.Error?

- ReadKey
- WriteKey
- ReadLine
- WriteLine

Choose the output return when read() reads the character from the console?

- Boolean
- String
- Char
- Integer

Choose the output for following set of code?



```
static void Main(string[] args)
{
Console.WriteLine("This is a Console Application:");
Console.Write("Please enter your lucky number:");
    string val1 = Console.ReadLine();
int val2 = System.Convert.ToInt32(val1, 10);
    val2 = val2 * val2;
Console.WriteLine("square of number is:" +val2);
Console.Read();
}
```

- (•) Run successfully ask for input and hence display the results
- ( ) Compile time error
- ( ) Run successfully donot prints anything
- ( ) Syntax Error

What would be the output for following input from the console as a character?

```
static void Main(string[] args)
{
Console.WriteLine("what is your name?");
    char s;
    s = Convert.ToChar(Console.ReadLine());
Console.WriteLine("how are you: "+s);
Console.Read();
}
```

- (•) Run time error
- ( ) Code run successfully prints nothing on console
- ( ) Compile time error
- ( ) Code run successfully prints input on console

Name the exception thrown by read() on failure.

- ( ) SystemInputException

Formatted: Font: Times New Roman
Formatted Table
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

- InterruptedException
- I/O Exception
- SystemException

Which of these methods are used to read single character from the console?

a) get()
b) getline()
c) read()
d) readLine()

Constructor

Please read the questions carefully and choose the most appropriate option.Which of given options is TRUE about constructors in C#.NET?

- A constructor cannot be overloaded.
- A constructor can be a static constructor.
- A constructor cannot be declared as private.
- None of the given options

What will be the output of given set of code?

```
class maths
{
 int i;
 public maths(int x)
 {
 i = x;
 Console.WriteLine("hello:");
 }
}
class maths1 : maths
{
 public maths1(int x) :base(x)
 {
 Console.WriteLine("bye");
 }
}
```

```
class Program
{
 static void Main(string[] args)
 {
 maths1 k = new maths1(12);
 Console.ReadLine();
 }
}
```

- bye

  12
- hello:

  bye
- 12

  hello
- Compile time error

What will be the output of given set of code?

```
 class maths
 {
 public int length;
 public int breadth;
 public maths(int x)
 {
 length = x + 1;
 }
 public maths(int x, int y)
 {
 length = x + 2;
 }
 }
 class Program
 {
 static void Main(string[] args)
 {
 maths m = new maths(6);
 maths k = new maths(6, 2);
 Console.WriteLine(m.length);
 Console.WriteLine(k.length);
 Console.ReadLine();
 }
 }
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt

- 7
  8
- 8
  8
- 8
  10
- 0
  2

What will be the output of given set of code?

```
class maths
{
public int length;
public int breadth;
public maths(int x, int y)
{
length = x;
breadth = y;
Console.WriteLine(x + y);
}
public maths(double x, int y)
{
length = (int)x;
breadth = y;
Console.WriteLine(x * y);
}
}
class Program
{
static void Main(string[] args)
{
maths m = new maths(20, 40);
maths k = new maths(12.0, 12);
Console.ReadLine();
}
}
```

- 60
  144.0
- 60

24

60

144

60

0

What will be the output of given set of code?

```
class maths
{
static maths()
{
int s = 8;
Console.WriteLine(s);
}
public maths(int f)
{
int h = 10;
Console.WriteLine(h);
}
}
class Program
{
static void Main(string[] args)
{
maths p = new maths(0);
Console.ReadLine();
}
}
```

8

10

10

10

8

8

0

10

What will be the output of given set of code?

```
class maths
{
   int i;
   public  maths(int ii)
   {
      ii = -25;
      int g;
      g = ii > 0 ? ii : ii * -1;
      Console.WriteLine(g);
   }
}
class maths1 :maths
{
   public  maths1(int ll) :base(ll)
   {
      ll = -1000;
      Console.WriteLine((ll > 0 ? ll : ll * -1));
   }
}
class Program
{
   static void Main(string[] args)
   {
      maths1 p = new maths1(6);
      Console.ReadLine();
   }
}
```

- -1025
- 25
- 1000
- -1025
- None of mentioned

What will be the output of given set of code?

```
class maths
{
   public maths()
   {
      Console.WriteLine("constructor 1 :");
   }
   public maths(int x)
   {
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
        int p = 2;
        int u;
        u = p + x;
        Console.WriteLine("constructor 2: " +u);
    }
}
class Program
{
    static void Main(string[] args)
    {
        maths k = new maths(4);
        maths t = new maths();
        Console.ReadLine();
    }
}
```

- constructor 1:

  constructor 2: 6
- None of the mentioned
- constructor 2: 6

  constructor 2: 6
- constructor 2: 6

  constructor 1:

What will be the output of given set of code?

```
class maths
{
int i;
public maths(int ii)
{
ii = 12;
int j = 12;
int r = ii * j;
Console.WriteLine(r);
}
}
class maths1 : maths
{
public maths1(int u) :base(u)
{
u = 13;
int h = 13;
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
Console.WriteLine(u + h);
}
}
class maths2 : maths1
{
public maths2(int k) :base(k)
{
k = 24;
int o = 6 ;
Console.WriteLine(k /o);
}
}
class Program
{
static void Main(string[] args)
{
maths2 t = new maths2(10);
Console.ReadLine();
}
}
```

- 26
  4
  144
- 4
  26
  144
- 144
  26
  4
- 0
  0
  0

Please read the questions carefully and choose the most appropriate option.How many times can a constr

- Only once
- As many times as we call it
- Twice

- Any number of times before the object gets garbage collected.

Correct statements about constructor overloading in C# is?

- Overloaded constructors have same name as the class name
- Overloaded constructors can use optional arguments
- Overloaded constructors can have different type of number of arguements as well as differ in number
- All the given options

Control structures

Select the output for following set of code :

```
static void Main(string[] args)
{
 int x = 0;
 do
 {
 x++;
 if (x == 5)
 {
 x++;
 continue;
 break;
 }
 Console.WriteLine(x + " ");
 }while (x < 10);
}
```

- 10
- 5 6 7 8 9 10
- 1 2 3 4 5
- 1 2 3 4 7 8 9 10

Select output for following set of code :

```
static void Main(string[] args)
{
    int i = 1, j = 2, k = 3;
    do
    {
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
    Console.WriteLine((Convert.ToBoolean(Convert.ToInt32(i++))) && (Convert.ToBoolean(Convert.ToInt32(++j))));
  }while (i <= 3);
  Console.ReadLine();
}
```

- 1 1 1
- 0 0 0
- True True True
- False False False

What will be the output for given set of code?

```
static void Main(string[] args)
{
int []arr = new int[]{ 1, 2, 3, 4, 5};
fun (ref arr);
for (int i = 0; i < arr.Length ; i++)
Console.WriteLine( arr[i] + " ");
}
static void fun(ref int[]a)
{
a = new int[6];
a[3] = 32;
a[1] = 24;
}
```

- 0

  0

  32

  0

  0

  0
- 0

  24

  0

  32

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

0

0

0

0

32

0

0

0

24

0

32

0

0

0

Please read the questions carefully and choose the most appropriate option.Which of the given options a
1.The goto statement passes control to the next iteration of the enclosing iteration statement in which it appears.
2.Branching is performed using jump statements which cause an immediate transfer of the program control.

Both 1 and 2

None of the listed options

only 2

only 1

What is output for following code snippet?

```
class Program
{
static void Main(string[] args)
{
int i = 5;
int j;
method1(ref i);
method2(out j);
Console.WriteLine(i + " " + j);
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
}
static void method1(ref int x)
{
x = x + x;
}
static void method2(out int x)
{
x = 6;
x = x * x;
}
}
```

- 36 10
- 10 36
- 0 0
- 36 0

Select the output for following set of Code:

```
static void Main(string[] args)
{
    int i;
    i = 0;
    while (i++ < 5)
    {
        Console.WriteLine(i);
    }
    Console.WriteLine("\n");
    i = 0;
    while ( ++i < 5)
   {
       Console.WriteLine(i);
   }
   Console.ReadLine();
}
```

- 1 2 3 4 5
- 1 2 3 4
- 1 2 3

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

1 2 3 4

1 2 3 4

1 2 3 4 5

1 2 3 4 5

Select the output for following set of code :

```
static void Main(string[] args)
{
 int x;
 for (x = 1; x <= 3; x++)
 {
 int j = 1;
 do
 {
 j++;
 }while (x % j == 2);
 Console.WriteLine(x + " " + j);
 }
 Console.ReadLine();
}
```

11

12

13

11

21

31

1 1

1 2

1 3

1 2

2 2

3 2

Which of these data types is used by operating system to manage the Recursion in Csharp?

- Array
- Tree
- Stack
- Queue

Select output for following set of code :

```
static void Main(string[] args)
{
    int i, s = 0, a = 1, d;
    i = Convert.ToInt32(Console.ReadLine());
    do
    {
        d = i % (2 * 4);
        s = s + d * a;
    }while ((Convert.ToInt32(i = i / (2 * 4))) != 0 && (Convert.ToBoolean(Convert.ToInt32((a) = (a * 10)))));
    Console.WriteLine(s);
    Console.ReadLine();
}
```

enter i = 342.

- It finds octal equivalent of i
- It finds binary equivalent of i
- It finds sum of digits of i
- It finds reverse of i

What will be the output for given set of code?

```
static void Main(string[] args)
        {
            int n = 1;
            method(n);
            Console.WriteLine(n);
            method1(ref n);
            Console.WriteLine(n);
            Console.ReadLine();
        }
```

```
static void method(int num)
{
   num += 20;
   Console.WriteLine(num);
}
static void method1(ref int num)
{
   num += 20;
   Console.WriteLine(num);
}
```

11

21

21

11

21

1

21

21

1

1

1

1

21

21

21

21

Select the output for following set of code :

```
static void Main(string[] args)
{
   long  x;
   x = Convert.ToInt32(Console.ReadLine());
   do
   {
      Console.WriteLine(x % 10);
```

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted Table

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman, 12 pt

```
    }while ((x = x / 10) != 0);
    Console.ReadLine();
}
```

enter x = 1234.

- number of digits present in x
- prints sum of digits of 'x'
- prints '1'
- prints reverse of x

Select output for following set of code :

```
static void Main(string[] args)
{
   float i = 1.0f, j = 0.05f;
   do
   {
      Console.WriteLine(i++ - ++j);
   }while (i < 2.0f && j <= 2.0f);
   Console.ReadLine();
}
```

- 0.05
- -0.05
- -0.04999995
- 0.95

Predict the output for following set of code :

```
static void Main(string[] args)
{
int x;
x = Convert.ToInt32(Console.ReadLine());
int c = 1;
while (c <= x)
{
if (c % 2 == 0)
{
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt

```
Console.WriteLine("Execute while " + c + "\t" + "time");
}
c++;
}
Console.ReadLine();
}
```

for x = 8.

- Execute while 2 time
  Execute while 3 time
  Execute while 4 time
  Execute while 5 time
- Execute while 2 time
  Execute while 4 time
  Execute while 6 time
  Execute while 8 time
- Execute while 1 time
  Execute while 2 time
  Execute while 3 time
  Execute while 4 time
  Execute while 5 time
  Execute while 6 time
  Execute while 7 time
- Execute while 1 time
  Execute while 3 time
  Execute while 5 time
  Execute while 7 time

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt

Select the output for following set of code:

```
static void Main(string[] args)
{
   float s = 0.1f;
   while (s <= 0.5f)
   {
```

```
        ++s;
        Console.WriteLine(s);
    }
    Console.ReadLine();
}
```

- 0.1 0.2 0.3 0.4 0.5
- 1.1
- 0.1
- No output

Select the output for following set of code:

```
static void Main(string[] args)
{
    float s = 0.1f;
    while (s <= 0.5f)
    {
        ++s;
        Console.WriteLine(s);
    }
    Console.ReadLine();
}
```

- 0.1 0.2 0.3 0.4 0.5
- 1.1
- 0.1
- No output

Please read the questions carefully and choose the most appropriate option.Which of the given options c r
1.break
2.goto

- only 1
- Both 1 and 2
- None of the listed options

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

- only 2

Which of these is not a correct statement?

- Recursion is always managed by C# Runtime environment
- Recursive methods are faster that programmers written loop to call the function repeatedly using a s
- A recursive method must have a base case
- Recursion always uses stack

What is Recursion in CSharp defined as?

- Recursion is a process of defining a method that calls other methods which in turn call again this m
- Recursion is a process of defining a method that calls itself repeatedly
- Recursion another form of class
- Recursion another process of defining a method that calls other methods repeatedly

Select the output for following set of Code:

```
static void Main(string[] args)
{
    int i;
    i = 0;
    while (i++ < 5)
    {
        Console.WriteLine(i);
    }
    Console.WriteLine("\n");
    i = 0;
    while ( ++i < 5)
    {
        Console.WriteLine(i);
    }
    Console.ReadLine();
}
```

- 1 2 3

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

1 2 3 4

1 2 3 4 5

1 2 3 4

1 2 3 4

1 2 3 4 5

1 2 3 4 5

1 2 3 4 5

1 2 3 4 5

Select output for following set of code :

```
static void Main(string[] args)
{
   int i = 1, j = 2, k = 3;
   do
   {
      Console.WriteLine((Convert.ToBoolean(Convert.ToInt32(i++))) && (Convert.ToBoolean(Convert.ToInt32(++j))));
   }while (i <= 3);
   Console.ReadLine();
}
```

1 1 1

0 0 0

True True True

False False False

Select the output for following set of Code:

```
static void Main(string[] args)
{
   int x = 0;
   while (x < 20)
   {
      while (x < 10)
      {
         if (x % 2 == 0)
         {
            Console.WriteLine(x);
         }
```

```
            x++;
        }
    }
    Console.ReadLine();
}
```

- (•) 0 2 4 6 8
- ( ) 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
- ( ) 0 2 4 6 8 10
- ( ) 0 2 4 6 8 10 12 14 16 18 20

aData types

Please read the questions carefully and choose the most appropriate option.Which of the given options are TRUE about enumerators?
1.The value of each successive enumerator is decreased by 1.
2.Values of enum elements cannot be populated from a database.

- ( ) None of the listed options
- ( ) only 1
- (•) only 2
- ( ) Both 1 and 2

What is output for following set of code ?

```
static void Main(string[] args)
{
 double a = 345.09;
 byte c = (byte) a;
Console.WriteLine(c);
Console.ReadLine();
}
```

- ( ) 88
- ( ) 84
- (•) 89
- ( ) 98

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt

Correct output for the C#.NET code given below is?

```
enum colors
      {
         red,
         black,
         pink
      }

      static void Main(string[] args)
      {
         colors s = colors.black;
         Type t;
         t = s.GetType();
         string[] str;
         str = Enum.GetNames(t);
         Console.WriteLine(str[0]);
         Console.ReadLine();
      }
```

- black
- 0
- red
- 1

Which of these method of class String is used to check whether a given string starts with a particular subs

- Ends()
- StartsWith()
- Starts()
- EndsWith()

What will be the output of give code snippet?

```
class Program
{
 static void Main(string[] args)
 {
 String s1 = "Hello i love Csharp";
StringBuilder s2 = new StringBuilder(s1);
Console.WriteLine(s1.Equals(s2));
Console.ReadLine();
 }
}
```

Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

- ( ) Compile time error
- ( ) true
- (•) False
- ( ) 0

Please read the questions carefully and choose the most appropriate option.Which of the given options ar
1.Anenum variable can be defined inside a class or a namespace.
2.Anenum variable cannot have a protected access modifier.

- (•) only 1
- ( ) None of the listed options
- ( ) only 2
- ( ) Both 1 and 2

Please read the questions carefully and choose the most appropriate option.Which of the given options ar

- ( ) A String is created on the stack.
- (•) A String is created on the heap.
- ( ) A String is a primitive.
- ( ) None of the given options

Choose the correct output for given set of code?

```
enum per
 {
     a,
     b,
     c,
     d,
 }
per.a = 10;
Console.writeline(per.b);
```

- ( ) 2
- (•) Compile time error
- ( ) 1
- ( ) 11

Choose the correct output for given set of code?

```
enumcolor:int
{
   red,
   green,
   blue = 5,
   cyan,
   pink = 10,
   brown
}
 static void Main(string[] args)
 {
      Console.WriteLine((int)color.green);
      Console.WriteLine((int)color.brown);
  }
```

- (•) 1 11
- ( ) 1 5
- ( ) 2 10
- ( ) 2 11

What will be the output of set of code?

```
static void Main(string[] args)
{
int [] a = {1, 2, 3, 4, 5};
   fun(a);
Console.ReadLine();
}
static void fun(paramsint[] b )
{
int[] k = { 3, 4, 7, 8,'\0' };
   for (inti = 0; i<b.Length; i++)
   {
      b[i] = b[i] + k[i] ;
Console.WriteLine( b[i] + " ");
   }
}
```

- (•) 4, 6, 10, 12, 5
- ( ) 3, 4, 7, 8, 5

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

- Compile time error
- 3, 4, 7, 8, 5, 1, 2, 3, 4, 5

Data types

Please read the questions carefully and choose the most appropriate option.Which of the given options are TRUE about a String?

- A String is created on the heap.
- None of the given options
- A String is a primitive.
- A String is created on the stack.

please read the questions carefully and choose the most appropriate option.Which of the given data type

- long
- int
- byte
- short

The modifiers used to define an array of parameters or lists of arguements:

- ref
- param
- var
- out

Choose the correct output for given set of code?

```
enum per
 {
    a,
    b,
    c,
    d,
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt

```
 }
per.a = 10;
Console.writeline(per.b);
```

○ 2

○ 11

● Compile time error

○ 1

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

What will be the output of given set of code?

```
 static void Main(string[] args)
 {
int[] x = { 80, 82, 65, 72, 83, 67 };
     fun(x);
Console.ReadLine();
 }
 static void fun(paramsint [] b )
 {
inti;
     for (i = 5; i>=0 ; i--)
     {
Console.WriteLine(Convert.ToChar(b[i]));
     }
 }
```

○ 67 83 72 65 82 80

○ 80 82 65 72 83 67

● C S H A R P

○ P R A H S C

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

Which of these method of class String is used to check whether a given string starts with a particular subs

○ EndsWith()

○ Starts()

● StartsWith()

○ Ends()

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

Choose correct statement about the C#.NET code given below?

```
enumcolor:byte
{
```

```
    yellow = 500,
    green = 1000,
    pink = 1300
}
```

- bytes value cannot be assigned to enum elements
- As valid range of byte exceeded the compiler will report an error
- enum elements should always take successive values
- enum must always be of int type

Please read the questions carefully and choose the most appropriate option.Which of the given options ar
1.String literals can contain any character literal including escape sequences.
2.Attempting to access a character that is outside the bounds of the string results in an IndexOutOfRange

- Both 1 and 2
- only 2
- only 1
- None of the listed options

hat will be the output for given set of code ?

```
 static void Main(string[] args)
 {
 object[] a = {"1", 4.0f, "harsh"};
 fun(a);
Console.ReadLine();
 }
 static void fun(params object[] b)
 {
 for (inti = 0; i<b.Length - 1; i++)
Console.WriteLine(b[i] + " ");
 }
```

- 1
  4.0
  harsh
- 1
  4
  harsh
- 1

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

4

hars

1

4

What will be the output of set of code?

```
static void Main(string[] args)
{
int [] a = {1, 2, 3, 4, 5};
   fun(a);
Console.ReadLine();
}
static void fun(paramsint[] b )
{
int[] k = { 3, 4, 7, 8,'\0' };
   for (inti = 0; i<b.Length; i++)
   {
      b[i] = b[i] + k[i] ;
Console.WriteLine( b[i] + " ");
   }
}
```

4, 6, 10, 12, 5

3, 4, 7, 8, 5

Compile time error

3, 4, 7, 8, 5, 1, 2, 3, 4, 5

Correct output for the C#.NET code given below is?

```
enum letters
      {
         a,
         b,
         c
      }

      static void Main(string[] args)
      {
         letters l;
         l = letters.a;
         Console.WriteLine(l);
         Console.ReadLine();
```

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman, 12 pt

Formatted Table

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman, 12 pt

}

- 0
- a
- -1
- letters.a

Please read the questions carefully and choose the most appropriate option.Which of the given options are TRUE?
1.A variable cannot be assigned to an enum element.
2.An enumerator contains white space in its name.
only 1

- None of the listed options
- only 2
- Both 1 and 2

Please read the questions carefully and choose the most appropriate option.Which of the given options
1.An implicit cast is needed to convert from enum type to an integral type.
2.Anenum variable cannot have a public access modifier.

- only 1
- only 2
- None of the listed options
- Both 1 and 2

DEBUG

If debug point is on a methodcall,_____will execute the entire method at a time and stops at the nextlin

- StepOut
- Break
- Step over
- Step In

hat is the shortcut key that is used to Start or resume execution of your code and then halts execution wh

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

- Ctrl-F5
- Ctrl-F9
- Ctrl-F10
- Ctrl-Shift-F5

What is the shortcut key that is used to execute remaining lines out from procedure?

- Shift-F5
- Shift-F11
- F11
- F5

What is the shortcut key that is used to set the execution point to the line of code you choose

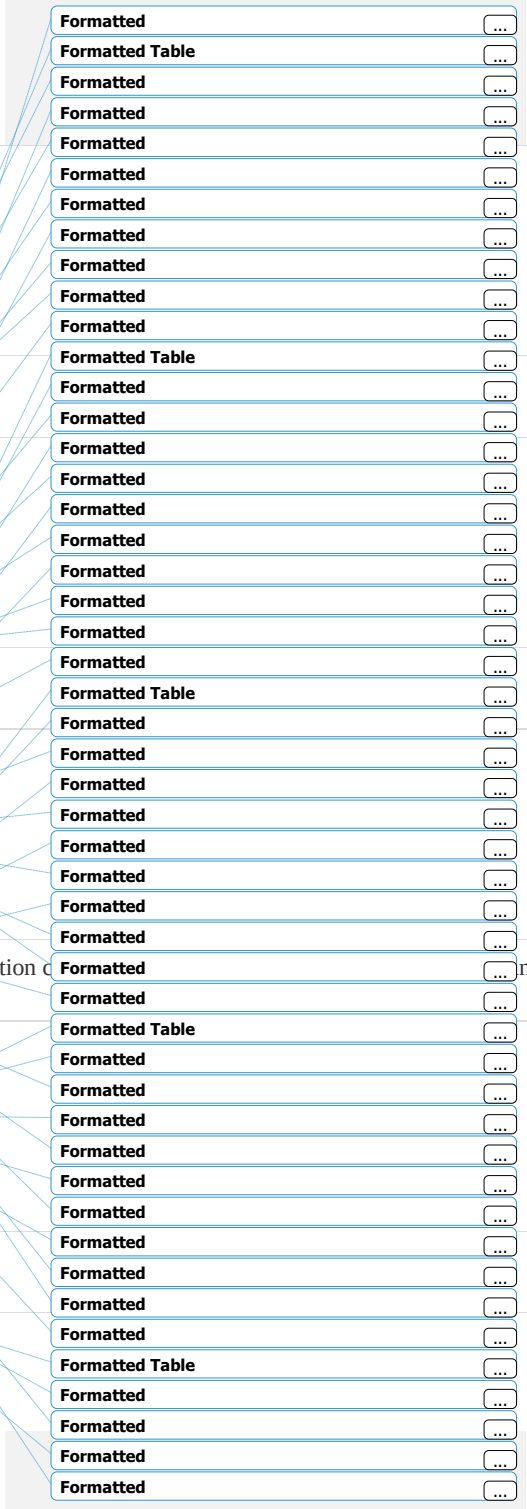
- Ctrl-Shift-F5
- Ctrl-F5
- Ctrl-F10
- Ctrl-Shift-F10

What is the shortcut key that is used to execute the next line of code but doesnot step into any function c

- Shift-F9
- Shift-F10
- Shift-F5
- Shift-F11

What is the shortcut key that is used to set or removes breakpoint at the current line?

- F5
- F6



- F10
- F9

What is the shortcut key that is used to allow you to attach or detach the debugger to one or more runnin

- Ctrl-Alt-H
- Ctrl-Alt-P
- Ctrl-Alt-W
- Ctrl-Alt-D

What is the shortcut key that is used to run the code without invoking debugger?

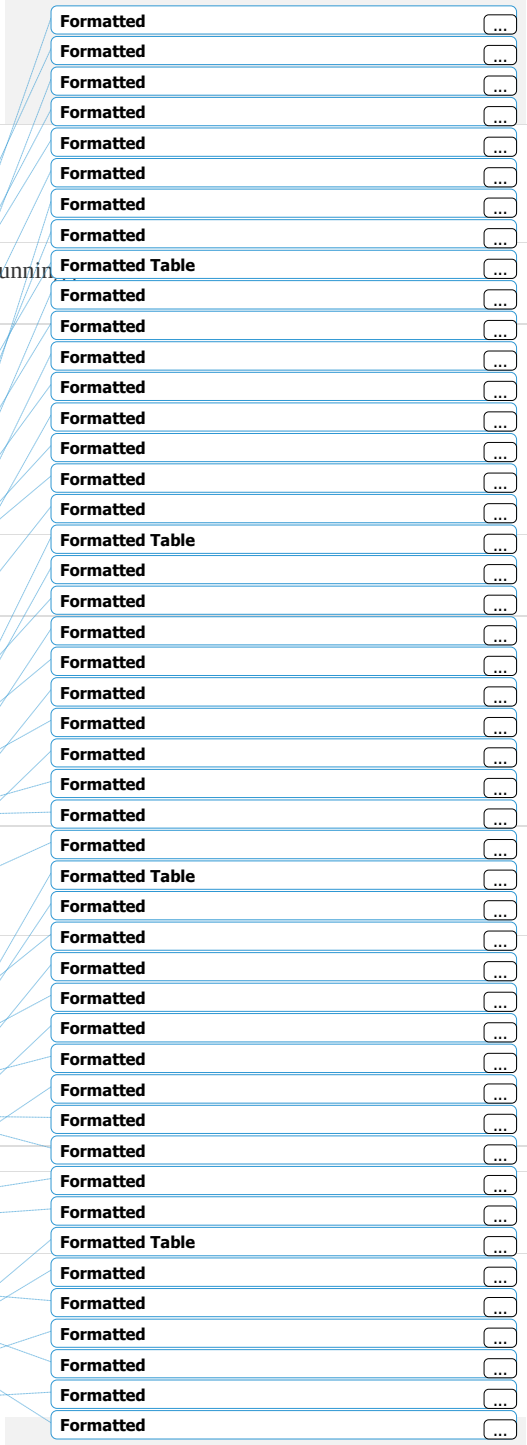
- F9
- F10
- Ctrl-F5
- F5

What is the shortcut key that is used to display the threads window to view all of the threads for the current process?

- Ctrl-Alt-P
- Ctrl-Alt-D
- Ctrl-Alt-W
- Ctrl-Alt-H

what are the commands that are not available in break mode to proceed for further debugging

- StepIn
- StepOut
- Continue



Break

GARBAGE COLLECTIONS

What will be the output of following snippet of code?

```
 class number
 {
int length = 50;
    public int number1
    {
       get
       {
          return length;
       }
       set
       {
          length = value;
       }
    }
 }
 class Program
 {
    public static void Main(string[] args)
    {
       number p = new number();
       p.number1 = p.number1 + 40;
int k = p.number1 * 3 / 9;
Console.WriteLine(k);
Console.ReadLine();
    }
 }
```

- 30
- 0
- 180
- Compile time error

Select the output for following set of Code :

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
 static void Main(string[] args)
 {
int a = 5, b = 10;
    if (Convert.ToBoolean(Convert.ToInt32(0xB)))
    if (Convert.ToBoolean(Convert.ToInt32(022)))
    if (Convert.ToBoolean(Convert.ToInt32('\xeb')))
Console.WriteLine("java");
    else ;
    else ;
    else ;
 }
```

- java
- Compile time error: Undefined symbol
- Compile time error: Misplaced else
- Warning: Condition is always true

Select output for set of code :

```
 static void Main(string[] args)
 {
int []a = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
func(ref a);
Console.ReadLine();
 }
 static void func(ref int[] x)
 {
Console.WriteLine(" numbers are:");
    for (inti = 0; i<x.Length; i++)
    {
       if (x[i] % 2 == 0)
       {
          x[i] = x[i] + 1;
Console.WriteLine(x[i]);
       }
    }
 }
```

- numbers are : 3 5 7 9 11
- None of the mentioned

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

- numbers are : 2 3 4 5 6
- numbers are : 2 4 6 8 10

Select the output for following set of code :

```
 static void Main(string[] args)
 {
int x = 8;
int b = 16;
int C = 64;
 x /= b /= C;
Console.WriteLine(x + " " + b+ " " +C);
Console.ReadLine();
 }
```

- 8 2 32
- Run time error
- 32 4 8
- 32 2 8

Select correct output for following set of code.

```
  static void Main(string[] args)
  {
int X = 0;
     if (Convert.ToBoolean(X = 0))
Console.WriteLine("It is zero");
     else
Console.WriteLine("It is not zero");
Console.ReadLine();
  }
```

- None of the mentioned
- It is not zero
- It is zero
- Infinite loop

Select the relevant ‘if statement’ to be placed in following set of code :

```
 static void Main(string[] args)
 {
int []num = {50, 65, 56, 88, 43, 52};
int even = 0, odd = 0;
    for (inti = 0 ;i<num.Length ;i++)
    {
        /*_____*/
    }
Console.WriteLine("Even Numbers:" +even);
Console.WriteLine("Odd Numbers:" +odd);
Console.ReadLine();
 }
```

```
if ((num % 2) == 0)
{
even += 1;
}
else
{
odd += 1;
}
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table

```
if(num[i] % 2 == 0)
{
even += 1;
}
else
{
odd += 1;
}
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

```
if((num * i) == 0)
{
even += 1;
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

```
    }
    else
    {
    odd += 1;
    }
    if(num[i] % 2 = 0)
    {
    even += 1;
    }
    else
    {
    odd += 1;
    }
```

Select the ouput for following set of code :

```
   static void Main(string[] args)
   {
int x = 4 ,b = 2;
      x -= b/= x * b;
Console.WriteLine(x + " " + b);
Console.ReadLine();
   }
```

- 4 0
- 0 4
- 4 2
- None of mentioned

Select the output for following set of code :

```
 static void Main(string[] args)
 {
inti, j;
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt

```
for (i = 2; i>= 0; i--)
{
for (j = 0; j <= 2; j++)
{
if (i == j)
{
Console.WriteLine("1");
}
else
{
Console.WriteLine("0");
}
}
Console.WriteLine("\n");
}
Console.ReadLine();
}
```

- 0 1 0
  1 0 0
  0 0 1
- 1 0 0
  0 1 0
  0 0 1
- 1 0 0
  0 0 1
  0 1 0
- 0 0 1
  0 1 0
  1 0 0

Please read the questions carefully and choose the most appropriate option.
Which of the following jobs are NOT performed by Garbage Collector?

- All the listed options
- Closing unclosed database collections.
- Freeing memory on the stack.

Closing unclosed files.

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: 12 pt

Formatted: Font: 12 pt

What will be the output of following snippet of code?

```
class number
{
private int num1 = 60;
private int num2 = 20;
public intanumber
{
get
{
return num1;
}
set
{
num1 = value;
}
}
public int anumber1
{
get
{
return num2;
}
set
{
num2 = value;
}
}
}
class Program
{
public static void Main(string[] args)
{
number p = new number();
number k = new number();
int m = p.anumber;
int t = k.anumber1;
int r = p.anumber * k.anumber1;
Console.WriteLine("sum = " + r);
Console.ReadLine();
}
}
```

- sum = 0
- sum = 120
- Compile time error
- sum = 1200

Please read the questions carefully and choose the most appropriate option.Imagine the scenario below. On pushing a button an object is to be notified, but it is not known until runtime which object should be Which of the given programming constructs should be used to implement this idea?

- Namespace
- Interface
- Delegate
- Attribute

Select the output for following set of Code :

```
 static void Main(string[] args)
 {
int a = -1;
int b = -1;
    if (Convert.ToBoolean (++a = ++b))
Console.WriteLine("a");
    else
Console.WriteLine("b");
Console.ReadLine();
 }
```

- b
- a
- compile time error
- Code execute successfully with no output

What will be the output of following snippet of code?

```
class number
{
```

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
int length = 60;
    public int number1
    {
       get
       {
          return length;
       }
    }
 }
 class Program
 {
    public static void Main(string[] args)
    {
       number p = new number();
int l;
       l = p.number1 + 40;
int k = l * 3 / 4;
Console.WriteLine(k);
Console.ReadLine();
    }
 }
```

- 80
- 0
- 75
- 30

What is the method to load assembly by name

- Assembly.loadfile()
- Assembly.reflectiononlyload
- Assembly.load from()
- Assembly.load()

What will be the output of following snippet of code?

```
class number
{
private int num1;
private int num2;
```

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
public intanumber
{
get
{
return num1;
}
set
{
num1 = value;
}
}
public int anumber1
{
get
{
return num2;
}
set
{
num2 = value;
}
}
}
class Program
{
public static void Main(string[] args)
{
number p = new number();
p.anumber = 20;
number k = new number();
k.anumber1 = 40;
int m = p.anumber;
int t = k.anumber1;
int r = p.anumber + k.anumber1;
Console.WriteLine("number = " +m);
Console.WriteLine("number = " +t);
Console.WriteLine("sum = " +r);
Console.ReadLine();
}
}
```

- None
- sum = 60

number = 40

Formatted: Font: Times New Roman

Formatted Table

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

number = 20

number = 20

number = 40

sum = 60

Compile time error

Linq

Assume 2 columns named as Product and Category how can be both sorted out based on first by category and then by product name?

var sortedProds = _db.Products.Orderby(c => c.Category) + ThenBy(n => n.Name)

var sortedProds = _db.Products.Orderby(c => c.Category)

All of the mentioned

var sortedProds = _db.Products.Orderby(c => c.Category) . ThenBy(n => n.Name)

For the given set of code what is output representing?

```
class Program
{
   static void Main(string[] args)
   {
      int[] nums = { 1, -2, 3, 0, -4, 5 };
      var posNums = from n in nums
                    where n > 0
                    select n;
      int len = posNums.Count();
      Console.WriteLine(len);
      Console.ReadLine();
   }
}
```

Execution of code with counting total numbers greater than zero

Execution of code with nothing being printed

Execution of code with printing all numbers

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman, 12 pt

Formatted Table

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman, 12 pt

Formatted Table

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

- Run time error

What will be the output of given code snippet?

```
class Program
{
   static void Main(string[] args)
   {
      int[] nums = { 1, -2, 3, 0, -4, 5};
      var posNums = from n in nums
                    where n >= 0
                    select n;
     foreach (int i in posNums)
     Console.Write(i + " ");
     Console.WriteLine();
     Console.ReadLine();
   }
}
```

- 1, 3, 0, 5
- 0, 1, -2, -4, 5
- 1, 3, 5
- Run time error

Choose the namespace in which Expression trees are encapsulated:

- System.Linq
- System.Text
- System.Linq.Expressions
- System.Collections.Generic

For the given set of code which query will work according to the set of code?

```
class Program
{
   static void Main(string[] args)
   {
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt

```
        int[] nums = { 1, -2, 3, 0, -4, 5 };
        int len = /*_____ */
        Console.WriteLine("The number of positive values in nums: " + len);
        Console.ReadLine();
    }
}
```

- from n in nums where n > 0
  select n
- (from n in nums where n > 0
  select n).Count();
- from n in nums where n > 0
  select n.Count()
- All of the mentioned

For the given set of code what is output?

```
class Program
{
    static void Main(string[] args)
    {
        int[] nums = { 1, -2, 3, 0, -4, 5 };
        var posNums = nums.Where(n => n < 10).Select(r => r%3);
        Console.Write("The values in nums: ");
        foreach (int i in posNums) Console.Write(i + " ");
        Console.WriteLine();
        Console.ReadLine();
    }
}
```

- Compile time error
- 2 -1 0 0 -2 1
- Run time error
- 1 -2 0 0 -1 2

Choose the namespace in which the interface IEnumerable is declared?

- B. System.Collections.Generic
- C. Both A and B
- D. None of the mentioned
- A. System.Collections

Select the output for given code snippet:

```
class Program
{
 static void Main(string[] args)
 {
 int[] nums = { 1, -2, 3, 0, -4, 5 };
 var posNums = from n in nums
 where n > -5 && n < 6
 orderby n descending
 select n;
 Console.Write("Descending order in nums: ");
 foreach (int i in posNums) Console.Write(i + " ");
 Console.WriteLine();
 Console.ReadLine();
 }
}
```

- Print nothing code run successfully
- Run time error
- Compile time error
- Arranged in descending order code run successfully

Select the namespace which should be included while making use of LINQ operations

- System.Collections.Generic
- None of the mentioned
- System.Text
- System.Linq

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt

For the given set of code what is output?

```
class Program
   {
      static void Main(string[] args)
      {

         int[] nums = { 1, -2, -3, 5 };
         var posNums = from n in nums
                       orderby n descending
                       select n*4 / 2;
         Console.Write("The values in nums: ");
         foreach (int i in posNums) Console.Write(i + " ");
         Console.WriteLine();
         Console.ReadLine();
      }
   }
```

- 10 2 -4 -6
- 5 1 -2 -3
- Run time error
- 1 5 -2 -3

Opps 4\

What is output following set of code ?

```
 using System;
 public class BaseClass
 {
    public BaseClass()
    {
Console.WriteLine("I am a base class");
    }
 }
 public class ChildClass : BaseClass
 {
    public ChildClass()
    {
Console.WriteLine ("I am a child class");
```

```
    }
 static void Main()
 {
ChildClass CC = new ChildClass();
 }
 }
```

○ None of the mentioned

○ I am a child class I am a base class

● I am a base class I am a child class

○ compile time error

Please read the questions carefully and choose the most appropriate option.Which of the given statemen

○ An interface can contain properties, methods and events.

○ Enhanced implementations of an interface can be developed without breaking existing code.

● All the listed options

○ Interfaces can be implemented by a class or a struct.

Please read the questions carefully and choose the most appropriate option.Which of the given statemen

○ A class that implements an interface can explicitly implement members of that interface

○ An interface can be implemented by multiple classes in the same program.

○ One interface can be implemented in another interface

○ The functions declared in an interface have a body

please read the questions carefully and choose the most appropriate option.Which of the given statement

○ Interfaces cannot be inherited

○ All the listed options

● Properties can be declared inside an interface

○ From two base interfaces a new interface cannot be inherited.

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

Please read the questions carefully and choose the most appropriate option.Which of the given statements is TRUE about an interface used in C#.NET?
1.One class can implement only one interface.
2.In a program if one class implements an interface then no other class in the same program can implement this interface.

- Only 1
- None of the listed options
- Only 2
- Both 1 and 2

Correct statement about following C#.NET code is?

```
class baseclass
  {
    int a;
    public baseclass(int a1)
    {
       a = a1;
      Console.WriteLine("a");
    }
    class derivedclass : baseclass
    {
       public derivedclass(int a1)
          : base(a1)
       {
          Console.WriteLine("b");
       }
    }
    class program
    {
       static void Main(string[] args)
       {
          derivedclass d = new derivedclass(20);
       }
    }
  }
```

- Output : b

  a
- the program will work correctly if we replace base(a1) with base.baseclass(a1)
- Output : a

b

- Compile time error

Select the output for given set of code?

```
public class sample
{
   public static int x = 100;
   public static int y = 150;

}
public class newspaper :sample
{
   new public static int x = 1000;
   static void Main(string[] args)
   {
console.writeline(sample.x + "  " + sample.y + "  " + x);
   }
}
```

- 100 150 100
- 1000 150 1000
- 100 150 1000
- 100 150 1000

Please read the questions carefully and choose the most appropriate option.
Which of the given statements is TRUE about an interface used in C#.NET?

- An interface can contain static data.
- If a class implements an interface partially, then it becomes an abstract class.
- An interface can contain static methods.
- A class cannot implement an interface partially.

Please read the questions carefully and choose the most appropriate option.Which of the given statemen
1.Interfaces can contain only method declaration.
2.Interfaces can contain static data and methods.

- only 2

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

- None of the listed options
- Both 1 and 2
- only 1

Please read the questions carefully and choose the most appropriate option.
It is possible to create a custom attribute that can be applied only to specific programming element(s) like which of the given options?

- Classes
- Methods
- Classes, Methods and Member-Variables
- Classes and Methods

Please read the questions carefully and choose the most appropriate option.Which of the given statemen
1.One class can implement only one interface.
2.In a program if one class implements an interface then no other class in the same program can implem

- Only 2
- None of the listed options
- Both 1 and 2
- Only 1

Which statement should be added in function a() of class y to get output “i love csharp”?

```
class x
   {
      public void a()
      {
         Console.WriteLine("i love csharp");
      }
   }
   class y : x
   {
      public void a()
      {
         /* add statement here */
         Console.Write("bye");
      }
   }
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
class program
{
   static void Main(string[] args)
   {
      y obj = new y();
      obj.a();
   }
}
```

- a()
- x.a();
- x::a();
- base.a();

Please read the questions carefully and choose the most appropriate option.Which of the given options c

1.Events
2.Structures

- None of the listed options
- only 2
- Both 1 and 2
- only 1

Please read the questions carefully and choose the most appropriate option.Which of the given statemen

- An interface cannot contain the signature of an indexer.
- Interfaces members are automatically public.
- When a class inherits an interface it inherits member definitions as well as its implementations.
- To implement an interface member, the corresponding member in the class must be public as well a

Please read the questions carefully and choose the most appropriate option.Which of the given options c

1.Properties
2.Method

- Both 1 and 2

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight

- None of the listed options
- only 1
- only 2

Please read the questions carefully and choose the most appropriate option.Which of the given options c
1.class
2.enum

- Both 1 and 2
- only 1
- None of the listed options
- only 2

Select statement added to the current set of code to get output as 10 20 ?

```
class baseclass
{
 protected int a = 20;
}
class derived : baseclass
{
int a = 10;
 public void math()
 {
 /* add code here */
 }
}
```

- Console.WriteLine(a + ” ” + base.a);
- Console.WriteLine(mybase.a + ” ” + a);
- Console.WriteLine(base.a + ” ” + a);
- Console.WriteLine(a + ” ” + this.a);

Oops 5

A class consists of two interfaces with each interface consisting of three methods.
The class had no instance data which indicates correct size of object created from this class?

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

- 12 bytes
- 16 bytes
- 0 bytes
- 24 bytes

Which of these statements is incorrect?

- Two thread in Csharp can have same priority
- Creating an instantiation for a thread doesn't mean that thread has started its execution process
- By multithreading CPU's idle time is minimized, and we can take maximum use of it
- A thread can exist only in two states, running and blocked

Which of these class is used to make a thread?

- String
- System
- Runnable
- Thread

The modifier used to define a class which does not have objects of it's own but acts as a base class for it

- abstract
- Static
- New
- Sealed

Given the class sample inherited by class sample 1. Which are correct statements about construction of object of class sample?

- The constructor of only sample class will be called
- While creating the object firstly the constructor of class sample will be called followed by constr
- While creating the object firstly constructor of class sample 1 will be called followed by construc
- The order of calling constructors depend on whether constructors in class sample and sample 1 ar

Formatted
Formatted Table
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted Table
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted Table
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted Table
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted

Select the class visibility modifiers among the following :

- Private, protected, public, internal
- Private, protected, public
- Private, protected, public, internal, protected internal
- All of the mentioned

Choose the correct output of following given code snippet?

```
interface i1
{
void f1();
}
interface i2 :i1
{
void f2();
}
public class maths :i2
{
public void f2()
{
Console.WriteLine("fun2");
}
public void f1()
{
Console.WriteLine("fun1");
}
}
class Program
{
static void Main()
{
maths m = new maths();
m.f1();
m.f2();
}
}
```

- fun1 fun2
- fun2 fun1
- fun2

Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

- fun1

Choose the correct statement about following code snippet given below:

```
interface a1
{
void f1();
void f2();
}
class a :a1
{
private int i;
void a1.f1()
{
}
}
```

- Compile time error
- Class a could not have an instance data
- Class a is an abstract class
- Class a fully implements the interface a1

What is multithreaded programming?

- Its a process in which a single process can access information from many sources
- It's a process in which two different processes run simultaneously
- It's a process in which two or more parts of same process run simultaneously
- Its a process in which many different process are able to access same information

Which of the following is the correct way of implementing an interface addition by class maths?

a) class maths : addition {}
b) class maths implements addition {}
c) class maths imports addition {}
d) None of the mentioned

Select the output for following set of codes:

```
static void Main(string[] args)
{
int i = 0;
while (i++ != 0) ;
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt

```
Console.WriteLine(i);
Console.ReadLine();
}
```

- It is not necessary to declare size of an array with it's type
- -127 to +127
- 1
- 0 to 127

A class member declared protected becomes member of subclass of which type?

- private member
- public member
- static member
- protected member

Select the output for following set of code:

```
static void Main(string[] args)
{
int i = 1, j = 1;
while (++i <= 10)
{
j++;
}
Console.WriteLine(i+ " " +j);
Console.ReadLine();
}
```

- It is not necessary to declare size of an array with it's type
- 11 10
- 12 11

Select the correct statement among the given statements?

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt

- Interfaces cannot be inherited
- Properties could be declared inside an interface
- None of the mentioned
- One class could implement only one interface

What will be the output of given code snippet?

```
interface calc
{
   void cal(int i);
}
class displayA :calc
{
   public int x;
   public void cal(int i)
   {
      x = i * i;
   }
}
class displayB :calc
{
   public int x;
   public void cal(int i)
   {
      x = i / i;
   }
}
class Program
{
   public static void Main(string[] args)
   {
      displayA arr1 = new displayA();
      displayB arr2 = new displayB();
      arr1.x = 0;
      arr2.x = 0;
      arr1.cal(2);
      arr2.cal(2);
      Console.WriteLine(arr1.x + " " + arr2.x);
      Console.ReadLine();
   }
}
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt

- 0 0
- 4 1
- 1 4
- 2 2

Oops 6

Correct way to implement the interface given below?

```
interface person
{
string firstname
{
get;
set;
}
}
```

- class emp :person{ private string str; public string firstname; { get { return str; } set { str = value; }
- class emp :implements person { private string str; public string firstname { get { return str; } set { s
- None of the mentioned
- class emp: implements person { private string str; public string person.firstname { get { return str; }

Correct code to be added for overloaded operator – for C# .net code given below?

```
 class csharp
 {
int x, y, z;
 public csharp()
 {

 }
 public csharp(int a ,int b ,int c)
 {
 x = a;
 y = b;
 z = c;
 }
 Add correct set of code here
 public void display()
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
 {
console.WriteLine(x + " " + y + " " + z);
 }
 class program
 {
 static void Main(String[] args)
 {
csharp s1 = new csharp(5 ,6 ,8);
csharp s3 = new csharp();
 s3 = - s1;
 s3.display();
 }
 }
}
```

- public static csharp operator -(csharp s1) { csharp t = new csharp(); t.x = s1.x; t.y = s1.y; t.z = s1.z;
- public static csharp operator -(csharp s1) { csharp t = new csharp(); t.x = -s1.x; t.y = -s1.y; t.z = -s1
- public static csharp operator -(csharp s1) { csharp t = new csharp(); t.x = s1.x; t.y = s1.y; t.z = -s1.z
- None of the mentioned

What will be the output for set of code?

```
 class maths
 {
    public int fun(int k, int y, int n)
    {
Console.WriteLine(k + "  " + y + "  " + n);
        return (k);
    }
    public int fun1(intt,float z)
    {
Console.WriteLine(t + "  " + z);
        return t;
    }
 }
 class Program
 {
    static void Main(string[] args)
    {
mathsobj = new maths();
int b = 90;
int c = 100;
int d ;
```

Formatted: Font: Times New Roman
Formatted Table
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: 12 pt
Formatted: Font: 12 pt

```
        float l;
inti = obj.fun(b, c, 12);
int j = (obj.fun1(12, 14.78f));
Console.ReadLine();
    }
 }
```

- 90, 100, 12 12, 14
- 0, 0, 0 12, 14.78
- 0, 0, 0 0, 0
- 90, 100, 12 12, 14.78

What would be output for set of code?

```
 class maths
 {
 public int x;
 public double y;
 public int add(int a, int b)
 {
 x = a + b;
 return x;
 }
 public int add(double c, double d)
 {
 y = c + d;
 return (int)y;
 }
 public maths()
 {
this.x = 0;
this.y = 0;
 }
 }
class Program
{
 static void Main(string[] args)
 {
mathsobj = new maths();
int a = 4;
 double b = 3.5;
obj.add(a, a);
obj.add(b, b);
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt

```
Console.WriteLine(obj.x + " " + obj.y);
Console.ReadLine();
 }
}
```

- 7.5 8
- 8 0
- 8 7
- 4 3.5

What will be the output of given code snippet?

```
 interface calc
 {
 void cal(inti);
 }
 public class maths :calc
 {
 public int x;
 public void cal(inti)
 {
 x = i * i;
 }
 }
 class Program
 {
 public static void Main(string[] args)
 {
mathsarr = new maths();
arr.x = 0;
arr.cal(2);
Console.WriteLine(arr.x);
Console.ReadLine();
 }
 }
```

- 0
- 2
- 4
- None of the mentioned

What could be the output of following set of code?

Formatted: Font: Times New Roman
Formatted Table
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Tab stops: 4.13", Left + Not at 0.64" + 1.27" + 1.91" + 2.54" + 3.18" + 3.82" + 4.45" + 5.09" + 5.73" + 6.36" + 7" + 7.63" + 8.27" + 8.91" + 9.54" + 10.18"
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
class Program
{
    static void Main(string[] args)
    {
Console.WriteLine( vol(10));
Console.WriteLine( vol(2.5f,  5));
Console.WriteLine( vol( 5l,  4,  5));
Console.ReadLine();
    }
    static intvol(int x)
    {
       return(x * x * x);
    }
    static float vol(float r,  int h)
    {
       return(3.14f * r * r * h);
    }
    static long vol(long l, int b, int h)
    {
       return(l * b * h);
    }
}
```

- 1000 0 100
- 0 0 100
- compile time error
- 1000 98.125 100

What will be the output for given set of code?

```
class maths
{
    public int fun(int ii)
    {
       return(ii > 0 ? ii :ii * -1);
    }
    public long fun(long ll)
    {
       return(ll> 0 ? ll :ll * -1);
    }
    public double fun( double dd)
    {
       return(dd> 0 ? dd :dd * -1);
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
    }
 }
 class Program
 {
    static void Main(string[] args)
    {
mathsobj = new maths();
inti = -25;
int j ;
        long l = -100000l ;
        long m;
        double d = -12.34;
        double e;
        j = obj.fun(i);
        m = obj.fun(l);
        e = obj.fun(d);
Console.WriteLine(j + "  " + m + "  " + e);
Console.ReadLine();
    }
 }
```

- 0 0 0
- 1 1 1
- 0
- 25 100000 12.34

Select the correct implementation of the interface which is mentioned below.

```
interface a1
{
int fun(inti);
}
```

- None of the mentioned
- class a: implements a1 { int fun(inti) { } }
- class a { int fun(inti) as a1.fun { } }
- class a: a1 { int a1.fun(inti) { } }

Select output for set of code?

```
 class sample
```

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
{
    public  inti;
    void display()
    {
Console.WriteLine(i);
    }
}
class sample1 : sample
{
   public  int j;
   public void display()
   {
Console.WriteLine(j);
   }
}
class Program
{
   static void Main(string[] args)
   {
      sample1 obj = new sample1();
obj.i = 1;
obj.j = 2;
obj.display();
Console.ReadLine();
   }
}
```

- 1
- 2
- 3
- Compile Time Error

The following set of code run on single level of inheritance. Find correct statement about the code?

```
 class sample
 {
inti = 10;
int j = 20;
    public void display()
    {
Console.WriteLine("base method ");
    }
 }
```

Formatted: Font: Times New Roman
Formatted Table
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
class sample1 : sample
{
   public  int s = 30;
}
class Program
{
   static void Main(string[] args)
   {
      sample1 obj = new sample1();
Console.WriteLine("{0}, {1}, {2}", obj.i,  obj.j,  obj.s);
obj.display();
Console.ReadLine();
   }
}
```

- compile time error
- base method
- 10, 20, 30 base method
- 10, 20, 0

Oops concept 1

Which of these can be used to fully abstract a class from its implementation?

- Packages
- Objects
- None of the Mentioned
- Interfaces

Access specifiers which can be used for an interface?

- Private
- All of the mentioned
- Protected

Formatted: Font: Times New Roman
Formatted Table
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

- (•) Public

Does C#.NET supports partial implementation of interfaces?

- ( ) true
- ( ) Can't Say
- (•) false
- ( ) None of the mentioned

Which of following keyword used to change data and behaviour of a base class by replacing a member
new derived member?

- ( ) Overrides
- (•) new
- ( ) Base
- ( ) Overloads

Which keyword used for correct implementation of an interface in C#.NET?

- ( ) intf
- ( ) Interface
- ( ) Intf
- (•) interface

Correct statement about C# code is?

```
public class maths
{
   public int x;
   public virtual void a()
   {

   }

}
public class subject : maths
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
{
   new public void a()
   {

   }

}
```

- subject class hides a() method of base class
- The code replaces the subject class version of a() with it's math class version
- The subject class version of a() method gets called using sample class reference which holds subject
- None of the mentioned

Which method is called when a thread is blocked from running temporarily?

- All of the mentioned
- Pulse()
- PulseAll()
- Wait()

Which of these keywords are used to implement synchronization?

- synchronized
- synchronize
- synch
- syn

Which statement correctly defines about Interfaces in C#.NET?

- Interfaces consists of only method declaration
- Interfaces cannot be inherited
- None of the mentioned

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight

- Interfaces consists of data static in nature and static methods

Choose the correct statements among the following:

- An abstract method does not have implementation
- An abstract method can take only either static or virtual modifiers
- An abstract method can be declared only in abstract class
- All of the mentioned

Oops concept 2

Please read the questions carefully and choose the most appropriate option.Which of the given options a

- C# does not support multiple inheritance in case of classes, but interfaces do support multiple inher
- In an interface all the methods are abstract.
- All of the listed options
- All interfaces should be declared with the keyword interface.

Please read the questions carefully and choose the most appropriate option.Which of the given options a

- None of the listed options
- Multiple classes may implement the same interface, and a single class may implement one or more
- Interfaces are essentially definitions of how a class needs to respond.
- In interface at least one method should not be abstract.

Please read the questions carefully and choose the most appropriate option.Is it possible to implement a

- Yes, but you should provide signatures of all methods, in the derived class
- No, there is a limit to the number of interfaces you can implement in a single derived class
- Yes, and you may or may not provide signature of all methods, in the derived class

Formatted
Formatted
Formatted
Formatted
Formatted Table
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted Table
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted Table
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted Table
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted
Formatted

Until the call of which type of method the newly created thread will not start executing?

- New()
- Start()
- All the given options
- Begin()

Please read the questions carefully and choose the most appropriate option.Which of the given options a

- We can use virtual method to provide default method implementation
- Both the listed options
- None of the 2 listed options
- We can use override keyword to change the implementation of the virtual methods in the sub class

Please read the questions carefully and choose the most appropriate option.Which of the given options a

- Child class instance cannot access the base class methods
- Child class instance can access the base class methods

Please read the questions carefully and choose the most appropriate option.Read the below statement ca
Statement 1: An interface in C# is a pure abstract class
Statement 2: An interface contains only definition of events, indexers, methods and/or properties.
Which of the above statements is TRUE about "Interfaces"?

- Both statements are true
- Only Statement 2 is true
- Only Statement 1 is true
- No Statement is true

Please read the questions carefully and choose the most appropriate option..
Which of the given options is TRUE about "Interfaces"?

- Classes and structs inheriting interfaces must provide an implementation for each interface mem
- Classes and structs inheriting interfaces, may or may not provide an implementation for each int

Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...

Please read the questions carefully and choose the most appropriate option.Inheritance enables you to c
in other classes?

- (•) Reuse, Extend and Modify
- ( ) Only Extend and Modify
- ( ) Only Reuse
- ( ) Only Modify

Please read the questions carefully and choose the most appropriate option.If you add a new method to a

- ( ) You have the option of providing default implementation and therefore all the existing code might
- (•) You have to track down all the implementations of the interface and define implementation for the
- ( ) None of the 2 listed options

Which statement correctly defines about Interfaces in C#.NET?

- (•) None
- ( ) Interfaces consists of data static in nature and static methods
- ( ) Interfaces consists of only method declaration
- ( ) Interfaces cannot be inherited

Please read the questions carefully and choose the most appropriate option.Which of the given options a
1.The subclass inherits all the super class attributes and extends them or adds others.
2.C Sharp supports multiple inheritance

- ( ) only 2
- ( ) Both 1 and 2
- (•) only 1
- ( ) None of the listed options

Please read the questions carefully and choose the most appropriate option.Read the below statements ca
Statement 1: Interface requires more time to find the actual method in the corresponding classes where a
Statement 2: Abstract method declarations are only permitted in abstract classes.
Which of the above statements are TRUE?

- (•) Only Statement 2 is true
- ( ) Both statements are true

- No Statement is true
- Only Statement 1 is true

Choose the correct statement about following code snippet in C#.NET:

```
interface abc
{
   String FirstName
   {
      get;
      set;
   }
  String LastName
  {
     get;
     set;
  }
  void print();
  void stock();
int fun();
}
```

- Properties cannot be declared inside an interface
- None of the mentioned
- Functions should be declared inside an interface
- It is workable code

Until the call of which type of method the newly created thread will not start executing?

- New()
- Begin()
- All the given options
- Start()

Choose the namespace which supports the multithreading programming:

- All of the mentioned
- System.net
- System.Linq
- System.Threading

Oops concept 3

Please read the questions carefully and choose the most appropriate option.Which of the given options are necessary for Run-time Polymorphism?

- Both the override method and the virtual method must have the same access level modifier.
- An abstract method is implicitly a virtual method.
- The overridden base method must be virtual, abstract or override.
- All the listed options

Please read the questions carefully and choose the most appropriate option.Which of the given options is

- When overriding a method, the names and type signatures of the override method must be the sa
- None of the 2 listed options
- Abstract methods are implicitly virtual.
- Both the listed options

What will be the output for given set of code?

```
class A
{
public virtual void display()
{
Console.WriteLine("A");
}
}
class B: A
{
public override void display()
{
Console.WriteLine("B");
}
}
class Program
{
static void Main(string[] args)
{
A obj1 = new A();
B obj2 = new B();
A r;
r = obj1;
r.display();
r = obj2;
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt

```
r.display();
Console.ReadLine();
 }
}
```

- A

  B
- Compile time error
- B

  B
- A

  A

Please read the questions carefully and choose the most appropriate option.Which of the given options is TRUE?

- None of the 2 listed options
- Both the listed options
- When used as a modifier, the new keyword explicitly hides a member inherited from a base cla
- Operator overloading works in different ways for structures and classes.

Please read the questions carefully and choose the most appropriate option.Which of the given options

- By default methods are virtual
- None of the 2 listed options
- If a derived class does not provide its own version of virtual method then the one in the base c
- Both the listed options

What will be the output for set of code?

```
 static void Main(string[] args)
 {
inti = 5;
int j  = 6;
    add(ref i);
    add(6);
Console.WriteLine(i);
Console.ReadLine();
 }
 static void add(ref int x)
```

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
{
   x = x * x;
}
static void add(int x)
{
Console.WriteLine(x * x * x);
}
```

- 25 0
- 216 0
- Compile time error
- 216 25

Please read the questions carefully and choose the most appropriate option.In which of the following sh

- Number of arguments
- Order of arguments
- Type of arguments
- All the listed options

Please read the questions carefully and choose the most appropriate option.Which of the given options

- We can use the new modifier to modify a nested type if the nested type is hiding another type.
- None of the 2 listed options
- Both the listed options
- Operator overloading permits the use of symbols to represent computations for a type.

Please read the questions carefully and choose the most appropriate option.Which of the following key

- operator
- All the listed options
- op

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

opoverload

Please read the questions carefully and choose the most appropriate option.Which of the given options

Operator overloading works in different ways for structures and classes.

Both the listed options

None of the 2 listed options

When used as a modifier, the new keyword explicitly hides a member inherited from a base class.

Please read the questions carefully and choose the most appropriate option.A derived class can stop vir

not inheritable

extends

sealed

inheritable

Please read the questions carefully and choose the most appropriate option.Which of the given options

The conditional logical operators cannot be overloaded.

Both the listed options

None of the 2 listed options

When a binary operator is overloaded the corresponding assignment operator, if any, must be expl

Please read the questions carefully and choose the most appropriate option.Which of the given options

&&

All the listed options

!

Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...

What will be the output for given set of code?

```
class a
{
public void fun()
{
Console.WriteLine("base method");
}
}
class b: a
{
public new void fun()
{
Console.WriteLine("derived method");
}
}
class Program
{
static void Main(string[] args)
{
b k = new b();
k.fun();
Console.ReadLine();
}
}
```

- Compile time error
- derived method
- base method
- Code run successfully print nothing

What will be the output for given set of code?

```
class maths
{
    public int fun(int k, int y)
    {
        return k + y;
    }
    public int fun1(int t, float z)
    {
        return (t+(int)z);
    }
```

```
}
class Program
{
   static void Main(string[] args)
   {
mathsobj = new maths();
inti;
int b = 90;
int c = 100;
int d = 12;
       float l = 14.78f;
i = obj.fun(b, c);
Console.WriteLine(i);
int j = (obj.fun1(d,  l));
Console.WriteLine(j);
Console.ReadLine();
   }
}
```

- 190, 0
- 190, 26.78f
- 190, 26
- 0, 26.78f

What will be the output for given set of code?

```
class a
{
public void fun()
{
Console.WriteLine("base method");
}
}
class b: a
{
public new void fun()
{
Console.WriteLine("derived method");
}
}
class Program
{
static void Main(string[] args)
```

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
 {
 b k = new b();
k.fun();
Console.ReadLine();
 }
 }
```

- derived method
- Compile time error
- base method
- Code run successfully print nothing

Please read the questions carefully and choose the most appropriate option.Which of the given options

- Events
- Methods
- Properties
- All the listed options

Please read the questions carefully and choose the most appropriate option.Which of the given options

- Both the listed options
- By default methods are virtual.
- None of the 2 listed options
- Each derived class does not have its own version of a virtual method.

What could be the output for set of code?

```
 class overload
 {
    public int x;
int y;
    public int add(int a)
    {
       x = a + 1;
       return x;
    }
```

Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

```
    public int add(int a, int b)
    {
       x = a + 2;
       return x;
    }
 }
 class Program
 {
    static void Main(string[] args)
    {
       overload obj = new overload();
       overload obj1 = new overload();
int a = 0;
obj.add(6);
       obj1.add(6, 2);
Console.WriteLine(obj.x);
Console.WriteLine(obj1.x);
Console.ReadLine();
    }
 }
```

- 8 8
- 8 10
- 7 8
- 0 2

Please read the questions carefully and choose the most appropriate option.Which of the given options

- None of the 2 listed options
- We can use the new modifier to modify a nested type if the nested type is hiding another type.
- Operator overloading permits the use of symbols to represent computations for a type.
- Both the listed options

Please read the questions carefully and choose the most appropriate option.Which of the given options

- None of the 2 listed options

Formatted: Font: Times New Roman
Formatted Table
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

- Both the listed options
- Abstract methods are implicitly virtual.
- When overriding a method, the names and type signatures of the override method must be the

Properties

choose the correct statement about properties describing the indexers?

- An indexer property should accept at least one argument
- No need to use the name of the property while using an indexed property
- Indexers can be overloaded
- All of the mentioned

Correct way to implement a write only property add in a math class?

- None
- class math

```
{
    public int add
    {
        set
        {
            add = value;
        }
    }
}
```

- class math

```
{
int ad;
    public int add
```

```
        {
            get
            {
                return ad;
            }
            set
            {
                ad = value;
            }
        }
    }
```

class math

```
    {
int ad;
        public int add
        {
            set
            {
                ad = value;
            }
        }
    }
```

Select the modifiers which can be used with the properties?

- All the given options
- Protected Internal
- Private
- Public
- Protected

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

Choose the correct statements about write-only properties in C#.NET?

- Properties which can only be set
- All of the listed options
- Properties once set and hence values cannot be read back in nature
- Useful for usage in classes which store sensitive information like password of a user

Consider a class maths and we had a property called as sum.b is a reference to a maths object and we want the statement Console.WriteLine(b.sum)to fail.
Which is the correct solution to ensure this functionality?

- Declare sum property with only get accessor
- Declare sum property with get, set and normal accessors
- Declare sum property with both get and set accessors
- Declare sum property with only set accessor

Correct way to implement a read only property add in a math class?

- None
-
```
class math
  {
    public int add
    {
      get
      {
        return ad;
      }
    }
  }
```
-
```
class math
  {
int ad;
```

Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt

```
        public int add
        {
            get
            {
                return ad;
            }
            set
            {
                ad = value;
            }
        }
    }
```

- class math

```
    {
int ad;
        public int add
        {
            get
            {
                return ad;
            }
        }


    }
```

Consider a class maths and we had a property called as sum.b is a reference to a maths object and we wa s

```
b.maths = 10;
Console.WriteLine(b.maths);
```

- Declare maths property with only set accessors

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman, 12 pt

Formatted: Font: Times New Roman, 12 pt

Formatted Table

Formatted: Font: Times New Roman

Formatted: Font: Times New Roman, 12 pt

- Declare maths property with only get, set and normal accessors
- Declare maths property with only get accessors
- Declare maths property with get and set accessors

If math class had add property with get accessors then which statements will work correctly?

- math m = new math();
  inti;
  i = m.add;
- math.add = 20;
- math m = new math();
  m.add = m.add + 20;
- math m = new math();
  m.add = 10;

Please read the questions carefully and choose the most appropriate option.A property can be declared i

- false
- true

A property can be declared inside a namespace or a procedure.

[A]. True [B].

Reflections

Which one of the following is an example of serialization

- All the given options

- [x] SOAP
- [ ] Session
- [ ] Database
- [ ] Diskfile

Which objects services multiple clients and hence it share data by storing state information between clie

- [x] Singleton
- [ ] Single Call
- [ ] Multiton
- [ ] Multi Call

What does the following code specifies?
object Invoke(object obj, object[] parameters)

- [ ] Calling a type using invoke()
- [ ] Any arguments that need to be passed to the method are specified in the array parameters
- [x] All the given options
- [ ] The value returned by the invoked method is returned by Invoke()

The property signifies “Obtains a Module object that represents the module (an executable file) in which
specifies the following statement:

- [ ] int MetadataToken
- [ ] Type DeclaringType
- [ ] Type ReflectedType
- [x] Module Module

Choose the namespace which consists of classes that are part of .NET Reflection API:

- [ ] None
- [ ] System.Name

Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...

- System.Text
- System.Reflection

How many types of serialization that are commonly used ?

- 5
- 3
- 2
- 4

What does the following method specifies?
Type[] GetGenericArguments()

- Obtains a list of the type arguments bound to a closed constructed generic type
- All the given options
- the list may contain both type arguments and type parameters
- a property defined by MemberInfo

DeSerialization is the process of converting _____ in to _____

- object to byte
- byte to array
- object to array
- byte to object

Which one of the following is not an advantage of XML Serialization?

- Greater Interoperability
- More Administrator Friendly
- All the given options
- Better Backward Compatibility

Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted Table ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...
Formatted ...

- Better Forward Compatability

Which Serialization is used for communicating between application that uses heterogeneous architecture

- Soap Serialization
- All the given options
- XML Serialization
- Binary Serialization

Xml

With XML:

- views are not limited to one multi-valued path only
- database data can automatically be extracted from XML documents only.
- All the given options
- documents can automatically be generated from database data only.

----------Interface

1. Which of the following can be declared in an interface?

1. Properties
2. Methods
3. Enumerations
4. Events
5. Structures

Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted Table
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt, Highlight
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt
Formatted: Font: Times New Roman, 12 pt

A. 1, 3

B. 1, 2, 4

C. 3, 5

D. 4, 5

Answer: Option B

2. A class implements two interfaces each containing three methods. The class contains no instance data. Which of the following correctly indicate the size of the object created from this class?

A. 12 bytes

B. 24 bytes

C. 0 byte

D. 8 bytes

E. 16 bytes

Answer: Option B

3. Which of the following statements is correct about an interface used in C#.NET?

A. One class can implement only one interface.

B. In a program if one class implements an interface then no other class in the same program can implement this interface.

C. From two base interfaces a new interface cannot be inherited.

D. Properties can be declared inside an interface.

E. Interfaces cannot be inherited.

Answer: Option D

4. Which of the following statements is correct about Interfaces used in C#.NET?

A. All interfaces are derived from an Object class.

B. Interfaces can be inherited.

C. All interfaces are derived from an Object interface.

D. Interfaces can contain only method declaration.

E. Interfaces can contain static data and methods.

Answer: Option B

5. Which of the following statements is correct about an interface used in C#.NET?

A. If a class implements an interface partially, then it becomes an abstract class.

B. A class cannot implement an interface partially.

C. An interface can contain static methods.

D. An interface can contain static data.

E. Multiple interface inheritance is not allowed.

Answer: Option A

6. Which of the following statements is correct about an interface?

A. One interface can be implemented in another interface.

B. An interface can be implemented by multiple classes in the same program.

C. A class that implements an interface can explicitly implement members of that interface.

D. The functions declared in an interface have a body.

Answer: Option C

7. Which of the following statements are correct about an interface in C#.NET?

1. A class can implement multiple interfaces.

2. Structures cannot inherit a class but can implement an interface.

3. In C#.NET, : is used to signify that a class member implements a specific interface.

4. An interface can implement multiple classes.

5. The static attribute can be used with a method that implements an interface declaration.

A. 1, 2, 3

B. 2, 4

C. 3, 5

D. None of the above.

Answer: Option A

8. Which of the following statements is correct?

A. When a class inherits an interface it inherits member definitions as well as its implementations.

B. An interface cannot contain the signature of an indexer.

C. Interfaces members are automatically public.

D. To implement an interface member, the corresponding member in the class must be public as well as static.

Answer: Option C

9. Which of the following statements are correct about an interface used in C#.NET?

1. An interface can contain properties, methods and events.

2. The keyword must implement forces implementation of an interface.

3. Interfaces can be overloaded.

4. Interfaces can be implemented by a class or a struct.

5. Enhanced implementations of an interface can be developed without breaking existing code.

A. 1, 2

B. 1, 4, 5

C. 3, 4

D. 3 only

Answer: Option B

10. Which of the following can implement an interface?

1. Data

2. Class

3. Enum

4. Structure

5. Namespace

A. 1, 3

B. 2, 4

C. 3, 5

D. 4 only

Answer: Option B

11. Which of the following unary operators can be overloaded?

1. true 2. false 3. + 4. new 5. is

A. 1, 2, 3

B. 3, 4, 5

C. 3 only

D. 5 only

Answer: Option A

12. A derived class can stop virtual inheritance by declaring an override as

A. Inherits

B. Extends

C. Inheritable

D. not inheritable

E. Sealed

Answer: Option E

13. Which of the following keyword is used to change the data and behavior of a base class by replacing a member of a base class with a new derived member?

A. new

B. base

C. overloads

D. override

E. overridable

Answer: Option A

14. Which of the following statements is correct?

A. When used as a modifier, the new keyword explicitly hides a member inherited from a base class.

B. Operator overloading works in different ways for structures and classes.

C. It is not necessary that all operator overloads are static methods of the class.

D. The cast operator can be overloaded.

Answer: Option A

15. Which of the following keyword is used to overload user-defined types by defining static member functions?

A. op

B. opoverload

C. operator

D. operatoroverload

E. udoperator

Answer: Option C

16. Which of the following statements is correct?

A. Static methods can be a virtual method.

B. Abstract methods can be a virtual method.

C. It is necessary to override a virtual method.

D. When overriding a method, the names and type signatures of the override method must be the same as the virtual method that is being overriden.

E. We can override virtual as well as non-virtual methods.

Answer: Option D

17. Which of the following statements are correct about a delegate?

1. Inheritance is a prerequisite for using delegates.

2. Delegates are type-safe.

3. Delegates provide wrappers for function pointers.

4. The declaration of a delegate must match the signature of the method that we intend to call using it.

5. Functions called using delegates are always late-bound.

A. 1 and 2 only

B. 1, 2 and 3 only

C. 2, 3 and 4 only

D. All of the above

E. None of the above

Answer: Option D

18. Which of the following statements are correct about delegates?

1. Delegates are not type-safe.

2. Delegate is a user-defined type.

3. Only one method can be bound with one delegate object.

4. Delegates can be used to implement callback notification.

5. Delegates permit execution of a method on a secondary thread in an asynchronous manner.

A. 1 and 2 only

B. 1, 2 and 3 only

C. 2, 4 and 5 only

D. 4 and 5 only

E. All of the above

Answer: Option C

19. Which of the following statements are correct about delegates?

A. Delegates cannot be used to call a static method of a class.

B. Delegates cannot be used to call procedures that receive variable number of arguments.

C. If signatures of two methods are same they can be called through the same delegate object.

D. Delegates cannot be used to call an instance function. Delegates cannot be used to call an instance subroutine.

Answer: Option B

20. Suppose on pushing a button an object is to be notified, but it is not known until runtime which object should be notified. Which of the following programming constructs should be used to implement this idea?

A. Attribute

B. Delegate

C. Namespace

D. Interface

E. Encapsulation

Answer: Option B

----------Delegate

21. Suppose a Generic class called SortObjects is to be made capable of sorting objects of any type (Integer, Single, Byte etc.). Which of the following programming constructs should be used to implement the comparision function?

A. Namespace B. Interface

C. Encapsulation D. Delegate E. Attribute

Answer: D

22. Which of the following statements is incorrect about a delegate?

A. A single delegate can invoke more than one method.

B. Delegates can be shared.

C. Delegate is a value type.

D. Delegates are type-safe wrappers for function pointers.

E. The signature of a delegate must match the signature of the method that is to be called using it.

Answer: Option C

23. With which of the following can the ref keyword be used?

1.Static data

2.Instance data

3.Static function/subroutine

4.Instance function/subroutine

A.1, 2

B.3, 4

C.1, 3

D.2, 4

E.All of the above

Answer: Option B

---

24. Which of the following statements are correct about an ArrayList collection that implements the IEnumerable interface?

1. The ArrayList class contains an inner class that implements the IEnumerator interface.

2. An ArrayList Collection cannot be accessed simultaneously by different threads.

3. The inner class of ArrayList can access ArrayList class's members.

4. To access members of ArrayList from the inner class, it is necessary to pass ArrayList class's reference to it.

5. Enumerator's of ArrayList Collection can manipulate the array.

A. 1 and 2 only

B. 1 and 3 and 4 only

C. 2 and 5 only

D. All of the above

E. None of the above

Answer: Option B

25. How many enumerators will exist if four threads are simultaneously working on an ArrayList object?

A. 1

B. 3

C. 2

D. 4

E. Depends upon the Project Setting made in Visual Studio.NET.

Answer: Option D

26. In which of the following collections is the Input/Output index-based?

1. Stack

2. Queue

3. BitArray

4. ArrayList

5. HashTable

A. 1 and 2 only

B. 3 and 4 only

C. 5 only

D. 1, 2 and 5 only

E. All of the above

Answer: Option B

27. Which of the following statements are correct about the Stack collection?

1. It can be used for evaluation of expressions.

2. All elements in the Stack collection can be accessed using an enumerator.

3. It is used to maintain a FIFO list.

4. All elements stored in a Stack collection must be of similar type.

5. Top-most element of the Stack collection can be accessed using the Peek() method.

A. 1 and 2 only

B. 3 and 4 only

C. 1, 2 and 5 only

D. All of the above

E. None of the above

Answer: Option C

28. A HashTable t maintains a collection of names of states and capital city of each state. Which of the following is the correct way to find out whether "Kerala" state is present in this collection or not?

A. t.ContainsKey("Kerala");

B. t.HasValue("Kerala");

C. t.HasKey("Kerala");

D. t.ContainsState("Kerala");

E. t.ContainsValue("Kerala");

Answer: Option A

29. Which of the following is NOT an interface declared in System.Collections namespace?

A. IComparer

B. Enumerable

C. Enumerator

D. IDictionaryComparer

E. IDictionaryEnumerator

Answer: Option D

30. Suppose value of the Capacity property of ArrayList Collection is set to 4. What will be the capacity of the Collection on adding fifth element to it?

A. 4

B. 8

C. 16

D. 32

[Dono]

31. Which of the following is an ordered collection class?

1. Map

2. Stack

3. Queue

4. BitArray

5. HashTable

A. 1 only

B. 2 and 3 only

C. 4 and 5 only

D. All of the above

E. None of the above

Answer: Option B

32. Which of the following is the correct way to find out the number of elements currently present in an ArrayList Collection called arr?

A. arr.Count

B. arr.GrowSize

C. arr.MaxIndex

D. arr.Capacity

E. arr.UpperBound

Answer: Option A

33. Which of the following statements are correct about a HashTable collection?

1. It is a keyed collection.

2. It is a ordered collection.

3. It is an indexed collection.

4. It implements aIDictionaryEnumerator interface in its inner class.

5. The key - value pairs present in a HashTable can be accessed using the Keys and Values properties of the inner class that implements the IDictionaryEnumerator interface.

A. 1 and 2 only

B. 1, 2 and 3 only

C. 4 and 5 only

D. 1, 4 and 5 only

E. All of the above

Answer: Option D

34. Which of the following statements are correct about the Collection Classes available in Framework Class Library?

A. Elements of a collection cannot be transmitted over a network.

B. Elements stored in a collection can be retrieved but cannot be modified.

C. It is not easy to adopt the existing Collection classes for newtype of objects.

D. Elements stored in a collection can be modified only if allelements are of similar types.

E. They use efficient algorithms to manage the collection, thereby improving the performance of the program.

Answer: Option E

35. Which of the following statements are correct?

1. Instance members of a class can be accessed only through an object of that class.

2. A class can contain only instance data and instance member function.

3. All objects created from a class will occupy equal number of bytes in memory.

4. A class can contain Friend functions.

5. A class is a blueprint or a template according to which objects are created.

A. 1, 3, 5

B. 2, 4

C. 3, 5

D. 2, 4, 5

E. None of these

Answer: Option A

36. Which of the following statements is correct?

A. Procedural Programming paradigm is different than structured programming paradigm.

B. Object Oriented Programming paradigm stresses on dividing the logic into smaller parts and writing procedures for each part.

C. Classes and objects are corner stones of structured programming paradigm.

D. Object Oriented Programming paradigm gives equal importance to data and the procedures that work on the data.

E. C#.NET is a structured programming language.

Answer: Option D

37. The this reference gets created when a member function (non-shared) of a class is called.

A. True B. False

Answer: Option A

38. Which of the following statements are correct?

1. Data membersofa class are by default public.

2. Data members of a class are by default private.

3. Member functions ofaclass are by default public.

4. A private function of a class can access a public function within the same class.

5. Member function of a class are by default private.

A. 1, 3, 5

B. 1, 4

C. 2, 4, 5

D. 1, 2, 3

E. None of these

Answer: Option C

39. Which of the following statements are correct about the this reference?

1. this reference can be modified in the instance member function of a class.

2. Static functions of a class never receive the this reference.

3. Instance member functions of a class always receive a this reference.

4. this reference continues to exist even after control returns from an instance member function.

5. While calling an instance member function we are not required to pass the this reference explicitly.

A. 1, 4

B. 2, 3, 5

C. 3, 4

D. 2, 5

E. None of these

Answer: Option B

40. Which of the following statements are correct about objects of a user-defined class called Sample?

1. All objects of Sample class will always have exactly same data.

2. Objects of Sample class may have same or different data.

3. Whether objects of Sample class will have same or different data depends upon a Project Setting made in Visual Studio.NET.

4. Conceptually, each object of Sample class will have instance data and instance member functions of the Sample class.

5. All objects of Sample class will share one copy of member functions.

A. 1, 3

B. 2, 4

C. 4, 5

D. 3, 5

E. None of these

Answer: Option C

41. Which of the following statements is correct about classes and objects in C#.NET?

A. Class is a value type.

B. Since objects are typically big in size, they are created on the stack.

C. Objects of smaller size are created on the heap.

D. Smaller objects that get created on the stack can be given names.

E. Objects are always nameless.

Answer: Option E

42. The [Serializable()] attribute gets inspected at

A. Compile-time

B. Run-time

C. Design-time

D. Linking-time

E. None of the above

Answer: Option B

43. Which of the following are correct ways to specify the targets for a custom attribute?

A. By applying AttributeUsage to the custom attribute's class definition.

B. By applying UsageAttribute to the custom attribute's class definition.

C. Once an attribute is declared it applies to all the targets.

D. By applying AttributeUsageAttribute to the custom attribute's class definition.

E. None of the above.

Answer: Option D

44. Which of the following are correct ways to pass a parameter to an attribute?

1. By value

2. By reference

3. By address

4. By position

5. By name

A. 1, 2

B. 1, 2, 3

C. 4, 5

D. All of the above

Answer: Option C

----------Attributes and C# .Net

45. Which of the following statements are correct about inspecting an attribute in C#.NET?

1. An attribute can be inspected at link-time.

2. An attribute can be inspected at compile-time.

3. An attribute can be inspected at run-time.

4. An attribute can be inspected at design-time.

A. 1, 2

B. 3, 4

C. 1, 3, 4

D. All of the above

E. None of the above

Answer: Option A

46. Which of the following statements are correct about Attributes used in C#.NET?

A. If there is a custom attribute BugFixAttribute then the compiler will look ONLY for the BugFix attribute in the code that uses this attribute.

B. To create a custom attribute we need to create a custom attribute structure and derive it from System.Attribute.

C. To create a custom attribute we need to create a class and implement IAttribute interface in it.

D. If a BugFixAttribute is to receive three parameters then the BugFixAttribute class should implement a zero-argument constructor.

E. The CLR can change the behaviour of the code depending upon the attributes applied to it.

Answer: Option E

47. Which of the following statements are correct about Attributes in C#.NET?

1. On compiling a C#.NET program the attibutes applied are recorded in the metadata of the assembly.

2. On compilation all the attribute's tags are deleted from the program.

3. It is not possible to create custom attributes..

4. The attributes applied can be read from an assembly using Reflection class.

5. An attribute can have parameters.

A. 1 and 2 only

B. 2 and 4 only

C. 1, 4 and 5 only

D. All of the above

E. None of the above

Answer: Option C

48. Which of the following correctly describes the contents of the filename AssemblyInfo.cs?

A. It contains method-level attributes.

B. It contains class-level attributes.

C. It contains assembly-level attributes.

D. It contains structure-level attributes.

E. It contains namespace-level attributes.

Answer: Option C

49. It possible to create a custom attribute that can be applied only to specific programming element(s) like ____

A. Classes

B. Methods

C. Classes and Methods

D. Classes, Methods and Member-Variables

Answer: Option C

---------- NameSpace and C#. NET

50. Which of the following CANNOT be a target for a custom attribute?

A. Enum

B. Event

C. Delegate

D. Interface

E. Namespace

Answer: Option E

51. Once applied which of the following CANNOT inspect the applied attribute?

A. CLR

B. Linker

C. ASP.NET Runtime

D. Visual Studio.NET

E. Language compilers

Answer: Option B

52. Attributes can be applied to

1. Method

2. Class

3. Assembly

4. Namespace

5. Enum

A. 1 and 2 only

B. 1, 2 and 3

C. 4 and 5 only

D. All of the above

E. None of the above

Answer: Option B.

53. If a namespace is present in a library then which of the following is the correct way to use the elements of the namespace?

A. Add Reference of the namespace. Use the elements of the namespace.

B. Add Reference of the namespace. Import the namespace. Use the elements of the namespace.

C. Import the namespace. Use the elements of the namespace.

D.Copy the library in the same directory as the project that is trying to use it. Use the elements of the namespace.

E. Install the namespace in Global Assembly Cache. Use the elements of the namespace.

Answer: Option B

54. Which of the following is NOT a namespace in the .NET Framework Class Library?

A. System.Process

B. System.Security

C. System.Threading

D. System.Drawing

E. System.Xml

Answer: Option A

55. Which of the following statements is correct about a namespace in C#.NET?

A. Namespaces help us to control the visibility of the elements present in it.

B. A namespace can contain a class but not another namespace.

C. If not mentioned, then the name 'root' gets assigned to the namespace.

D. It is necessary to use the using statement to be able to use an element of a namespace.

E. We need to organise the classes declared in Framework Class Library into different namespaces.

Answer: Option A

56. Which of the following is absolutely neccessary to use a class Point present in namespace Graph stored in library?

A. Use fully qualified name of the Point class.

B. Use using statement before using the Point class.

C. Add Reference of the library before using the Point class.

D. Use using statement before using the Point class.

E. Copy the library into the current project directory before using the Point class.

Answer: Option C

57. Which of the followings are NOT a .NET namespace?

1. System.Web

2. System.Process

3. System.Data

4. System.Drawing2D

5. System.Drawing3D

A. 1, 3

B. 2, 4, 5

C. 3, 5

D. 1, 2, 3

Answer: Option B

58. Which of the following statements is correct about namespaces in C#.NET?

A. Namespaces can be nested only up to level 5.

B. A namespace cannot be nested.

C. There is no limit on the number of levels while nesting namespaces.

D. If namespaces are nested, then it is necessary to use using statement while using the elements of the inner namespace.

E. Nesting of namespaces is permitted, provided all the inner namespaces are declared in the same file.

Answer: Option C

---------- using statement in C# .NET

59. Which of the following statements is correct about the using statement used in C#.NET?

A. using statement can be placed anywhere in the C#.NET source code file.

B. It is permitted to define a member at namespace level as a using alias.

C. A C#.NET source code file can contain any number of using statement.

D. By using using statement it is possible to create an alias for the namespace but not for the namespace element.

E. By using using statement it is possible to create an alias for the namespace element but not for the namespace.

Answer: Option C

---------- Namespace used in C# .NET

60. Which of the following statements are correct about a namespace used in C#.NET?

1. Classes must belong to a namespace, whereas structures need not.

2. Every class, struct, enum, delegate and interlace has to belong to some or the other namespace.

3. All elements of the namespace have to belong to one file.

4. If not mentioned, a namespace takes the name of the current project.

5. The namespace should be imported to be able to use the elements in it.

A. 1, 3

B. 2, 4, 5

C. 3, 5

D. 4 only

Answer: Option B

61. Which of the following CANNOT belong to a C#.NET Namespace?

A. class

B. struct

C. enum

D. Data

E. interface

Answer: Option D

62. Which of the following statements is correct about a namespace used in C#.NET?

A. Nested namespaces are not allowed.

B. Importing outer namespace imports inner namespace.

C. Nested namespaces are allowed.

D. If nested, the namespaces cannot be split across files.

Answer: Option C

---------- Property in C# .NET

63. A property can be declared inside a class, struct, Interface.

A. True B. False

Answer: Option A

64. Which of the following statements is correct about properties used in C#.NET?

A. A property can simultaneously be read only or write only.

B. A property can be either read only or write only.

C. A write only property will have only get accessor.

D. A write only property will always return a value.

Answer: Option B

65. A Student class has a property called rollNo and stu is a reference to a Student object and we want the statement stu.RollNo = 28 to fail. Which of the following options will ensure this functionality?

A. Declare rollNo property with both get and set accessors.

B. Declare rollNo property with only set accessor.

C. Declare rollNo property with get, set and normal accessors.

D. Declare rollNo property with only get accessor.

E. None of the above

Answer: Option D

66. Which of the following statements are correct?

1. The signature of an indexer consists of the number and types of its formal parameters.

2. Indexers are similar to properties except that their accessors take parameters.

3. Accessors of interface indexers use modifiers.

4. The type of an indexer and the type of its parameters must be at least as accessible as the indexer itself.

5. An interface accessor contains a body.

A. 1, 3, 5

B. 1, 2, 4

C. 3, 5

D. 2, 4

Answer: Option B

67. A property can be declared inside a namespace or a procedure.

A. True B. False

Answer: Option B

68. Which of the following statements is correct about properties used in C#.NET?

A. Every property must have a set accessor and a get accessor.

B. Properties cannot be overloaded.

C. Properties of a class are actually methods that work like data members.

D. A property has to be either read only or a write only.

Answer: Option C

69. Which of the folowing does an indexer allow to index in the same way as an array?

1. A class

2. A property

3. A struct

4. A function

5. An interface

A. 1, 3, 5

B. 2, 4

C. 3, 5

D. 3, 4, 5

Answer: Option A

70. An Employee class has a property called age and emp is reference to a Employee object and we want the statement Console.WriteLine(emp.age) to fail. Which of the following options will ensure this functionality?

A. Declare age property with only get accessor.

B. Declare age property with only set accessor.

C. Declare age property with both get and set accessors.

D. Declare age property with get, set and normal accessors.

E. None of the above

Answer: Option B

---------- enum /enumerator in C# .NET

71. Which of the following statements are correct about an enum used inC#.NET?

1. By default the first enumerator has the value equal to the number of elements present in the list.

2. The value of each successive enumerator is decreased by 1.

3. An enumerator contains white space in its name.

4. A variable cannot be assigned to an enum element.

5. Values of enum elements cannot be populated from a database.

A. 1, 2

B. 3, 4

C. 4, 5

D. 1, 4

Answer: Option C

72. Which of the following statements is true about an enum used in C#.NET?

A. An implicit cast is needed to convert from enum type to an integral type.

B. An enum variable cannot have a public access modifier.

C. An enum variable cannot have a private access modifier.

D. An enum variable can be defined inside a class or a namespace.

E. An enum variable cannot have a protected access modifier.

Answer: Option D

73. Which of the following statements are correct about an enum used inC#.NET?

1. To use the keyword enum, we should either use [enum] or System.Enum.

2. enum is a keyword.

3. Enum is class declared in System.Type namespace.

4. Enum is a class declared in the current project's root namespace.

5. Enum is a class declared in System namespace.

A. 1, 3

B. 2, 4

C. 2, 5

D. 3, 4

Answer: Option C

74. An enum that is declared inside a class, struct, namespace or interface is treated as public.

A. True B. False

Answer: Option A

75. An enum can be declared inside a class, struct, namespace or interface.

A. True B. False

Answer: Option A

76. Which of the following CANNOT be used as an underlying datatype for an enum in C#.NET?

A. byte

B. short

C. float

D. int

Answer: Option C

77. Which of the following statements are correct about enum used in C#.NET?

1. Every enum is derived from an Object class.

2. Every enum is a value type.

3. There does not exist a way to print an element of an enum as a string.

4. Every enum is a reference type.

5. The default underlying datatype of an enum is int.

A. 1, 2, 5

B. 1, 4

C. 3, 5

D. 2, 3, 4

Answer: Option A

78. Which of the following statements is correct about an enum used in C#.NET?

A. enum is a reference type.

B. enum is a value type.

C. Whether it a value type or a reference type depends upon size.

D. Whether it a value type or a reference type depends upon a Project Setting made in Visual Stiiclio.NET.

E. We can programmatically control whether it is a value type or a reference type.

Answer: Option B

79. Which of the following statements are correct about an enum used in C#.NET?

1. An enum can be declared inside a class.

2. An enum can take Single, Double or Decimal values.

3. An enum can be declared outside a class.

4. An enum can be declared inside/outside a namespace.

5. An object can be assigned to an enum variable.

A. 1, 3, 4

B. 2, 5

C. 3, 4

D. 2, 4, 5

Answer: Option A

---------- Strngs / Arrays in C.NET

80. The string built using the String class are immutable (unchangeable), whereas, the ones built- using the StringBuilder class are mutable.

A. True B. False

Answer: Option A

80.11. Which of the following statements about a String is correct?

A. A String is created on the stack.

B. Whether a String is created on the stack or the heap depends on the length of the String.

C. A String is a primitive.

D. A String can be created by using the statement String s1 = new String;

E. A String is created on the heap.

Answer: Option E

81. Which of the following statement is correct about a String in C#.NET?

A. A String is mutable because it can be modified once it has been created.

B. Methods of the String class can be used to modify the string.

C. A number CANNOT be represented in the form of a String.

D. A String has a zero-based index.

E. The System.Array class is used to represent a string.

Answer: Option D

82. Which of the following statements are correct about the String Class in C#.NET?

1. Two strings can be concatenated by using an expression of the form s3 = s1 + s2;

2. String is a primitive in C#.NET.

3. A string built using StringBuilder Class is Mutable.

4. A string built using String Class is Immutable.

5. Two strings can be concatenated by using an expression of the form s3 = s1&s2;

A. 1, 2, 5

B. 2, 4

C. 1, 3, 4

D. 3, 5

Answer: Option C

83. Which of the following statements are correct?

1. String is a value type.

2. String literals can contain any character literal including escape sequences.

3. The equality operators are defined to compare the values of string objects as well as references.

4. Attempting to access a character that is outside the bounds of the string results in an IndexOutOfRangeException.

5. The contents of a string object can be changed after the object is created.

A. 1, 3

B. 3, 5

C. 2, 4

D. 1, 2, 4

Answer: Option C

---------- Arrays In C# .NET

84. Which one of the following statements is correct?

A. Array elements can be of integer type only.

B. The rank of an Array is the total number of elements it can contain.

C. The length of an Array is the number of dimensions in the Array.

D. The default value of numeric array elements is zero.

E. The Array elements are guaranteed to be sorted.

Answer: Option D

85. Which of the following statements are correct about arrays used in C#.NET?

1. Arrays can be rectangular or jagged.

2. Rectangular arrays have similar rows stored in adjacent memory locations.

3. Jagged arrays do not have an access to the methods of System.Array Class.

4. Rectangular arrays do not have an access to the methods of System.Array Class.

5. Jagged arrays have dissimilar rows stored in non-adjacent memory locations.

A. 1, 2

B. 1, 3, 5

C. 3, 4

D. 1, 2, 5

E. 4, 5

Answer: Option D

---------- Structure in C# .NET

86. The space required for structure variables is allocated on stack.

A. True B. False

Answer: Option A

87. Creating empty structures is allowed in C#.NET.

A. True B. False

Answer: Option B

88. Which of the following statements are correct?

1. A struct can contain properties.

2. A struct can contain constructors.

3. A struct can contain protected data members.

4. A struct cannot contain methods.

5. A struct cannot contain constants.

A. 1, 2

B. 3, 4

C. 1, 2, 4

D. 3, 5

Answer: Option A

89. C#.NET structures are always value types.

A. True B. False

Answer: Option A

90. When would a structure variable get destroyed?

A. When no reference refers to it, it will get garbage collected.

B. Depends upon whether it is created using new or without using new.

C. When it goes out of scope.

D. Depends upon the Project Settings made in Visual Studio.NET.

E. Depends upon whether we free it's memory using free() or delete().

Answer: Option C

91. Which of the following statements is correct?

A. A struct never declares a default constructor.

B. All value types in C# inherently derive from ValueType, which inherits from Object.

C. A struct never declares a default destructor.

D. In C#, classes and structs are semantically same.

Answer: Option B

92. Which of the following are true about classes and struct?

1. A class is a reference type, whereas a struct is a value type.

2. Objects are created using new, whereas structure variables can be created either using new or without using new.

3. A structure variable will always be created slower than an object.

4. A structure variable will die when it goes out of scope.

5. An object will die when it goes out of scope.

A. 1, 2, 4

B. 3, 5

C. 2, 4

D. 3, 4, 5

Answer: Option A

93. Which of the following statements are correct about Structures used in C#.NET?

1. A Structure can be declared within a procedure.

2. Structs can implement an interface but they cannot inherit from another struct.

3. struct members cannot be declared as protected.

4. A Structure can be empty.

5. It is an error to initialize an instance field in a struct.

A. 1, 2, 4

B. 2, 3, 5

C. 2, 4

D. 1, 3

Answer: Option B

94. Which of the following statements are TRUE about the .NET CLR?

1. It provides a language-neutral development & execution environment.

2. It ensures that an application would not be able to access memory that it is not authorized to access.

3. It provides services to run "managed" applications.

4. The resources are garbage collected.

5. It provides services to run "unmanaged" applications.

A. Only 1 and 2

B. Only 1, 2 and 4

C. 1, 2, 3, 4

D. Only 4 and 5

E. Only 3 and 4

Answer: Option C

95. Which of the following are valid .NET CLR JIT performance counters?

1. Total memory used for JIT compilation

2. Average memory used for JIT compilation

3. Number of methods that failed to compile with the standard JIT

4. Percentage of processor time spent performing JIT compilation

5. Percentage of memory currently dedicated for JIT compilation

A. 1, 5

B. 3, 4

C. 1, 2

D. 4, 5

Answer: Option B

96. Which of the following statements is correct about Managed Code?

A. Managed code is the code that is compiled by the JIT compilers.

B. Managed code is the code where resources are Garbage Collected.

C. Managed code is the code that runs on top of Windows.

D. Managed code is the code that is written to target the services of the CLR.

E. Managed code is the code that can run on top of Linux.

Answer: Option D

97. Which of the following utilities can be used to compile managed assemblies into processor-specific native code?

A. gacutil B. ngen C. sn D. dumpbin E. ildasm

Answer: Option B

98. Which of the following are NOT true about .NET Framework?

1. It provides a consistent object-oriented programming environment whether object code is stored and executed locally, executed locally but Internet-distributed, or executed remotely.

2. It provides a code-execution environment that minimizes software deployment and versioning conflicts.

3. It provides a code-execution environment that promotes safe execution of code, including code created by an unknown or semi-trusted third party.

4. It provides different programming models for Windows-based applications and Web-based applications.

5. It provides an event driven programming model for building Windows Device Drivers.

A. 1, 2

B. 2, 4

C. 4, 5

D. 1, 2, 4

Answer: Option C

99. Which of the following components of the .NET framework provide an extensible set of classes that can be used by any .NET compliant programming language?

A. .NET class libraries

B. Common Language Runtime

C. Common Language Infrastructure

D. Component Object Model

E. Common Type System

Answer: Option A

100. Which of the following jobs are NOT performed by Garbage Collector?

1. Freeing memory on the stack.

2. Avoiding memory leaks.

3. Freeing memory occupied by unreferenced objects.

4. Closing unclosed database collections.

5. Closing unclosed files.

A. 1, 2, 3

B. 3, 5

C. 1, 4, 5

D. 3, 4

Answer: Option C

101. Which of the following .NET components can be used to remove unused references from the managed heap?

A. Common Language Infrastructure

B. CLR

C. Garbage Collector

D. Class Loader

E. CTS

Answer: Option C

102. Which of the following statements correctly define .NET Framework?

A. It is an environment for developing, building, deploying and executing Desktop Applications, Web Applications and Web Services.

B. It is an environment for developing, building, deploying and executing only Web Applications.

C. It is an environment for developing, building, deploying and executing Distributed Applications.

D. It is an environment for developing, building, deploying and executing Web Services.

E. It is an environment for development and execution of Windows applications.

Answer: Option A

103. Which of the following constitutes the .NET Framework?

1. ASP.NET Applications

2. CLR

3. Framework Class Library

4. WinForm Applications

5. Windows Services

A. 1, 2

B. 2, 3

C. 3, 4

D. 2, 5

Answer: Option B

104. Which of the following assemblies can be stored in Global Assembly Cache?

A. Private Assemblies

B. Friend Assemblies

C. Shared Assemblies

D. Public Assemblies

E. Protected Assemblies

Answer: Option C

105. Code that targets the Common Language Runtime is known as

A. Unmanaged

B. Distributed

C. Legacy

D. Managed Code

E. Native Code

Answer: Option D

106. Which of the following statements is correct about the .NET Framework?

A. .NET Framework uses DCOM for achieving language interoperability.

B. .NET Framework is built on the DCOM technology.

C. .NET Framework uses DCOM for making transition between managed and unmanaged code.

D. .NET Framework uses DCOM for creating unmanaged applications.

E. .NET Framework uses COM+ services while creating Distributed Applications.

Answer: Option C

107. Which of the following is the root of the .NET type hierarchy?

A. System.Object

B. System.Type

C. System.Base

D. System.Parent

E. System.Root

Answer: Option A

108. Which of the following benefits do we get on running managed code under CLR?

1. Type safety of the code running under CLR is assured.

2. It is ensured that an application would not access the memory that it is not authorized to access.

3. It launches separate process for every application running under it.

4. The resources are Garbage collected.

A. Only 1 and 2

B. Only 2, 3 and 4

C. Only 1, 2 and 4

D. Only 4

E. All of the above

Answer: Option E

109. Which of the following security features can .NET applications avail?

1. PIN Security

2. Code Access Security

3. Role Based Security

4. Authentication Security

5. Biorhythm Security

A. 1, 4, 5

B. 2, 5

C. 2, 3

D. 3, 4

Answer: Option C

110. Which of the following jobs are done by Common Language Runtime?

1. It provides core services such as memory management, thread management, and remoting.

2. It enforces strict type safety.

3. It provides Code Access Security.

4. It provides Garbage Collection Services.

A. Only 1 and 2

B. Only 3, 4

C. Only 1, 3 and 4

D. Only 2, 3 and 4

E. All of the above

Answer: Option E

111. Which of the following statements are correct about a .NET Assembly?

1. It is the smallest deployable unit.

2. Each assembly has only one entry point - Main(), WinMain() or DLLMain().

3. An assembly can be a Shared assembly or a Private assembly.

4. An assembly can contain only code and data.

5. An assembly is always in the form of an EXE file.

A. 1, 2, 3

B. 2, 4, 5

C. 1, 3, 5

D. 1, 2

Answer: Option A

112. Which of the following statements are correct about JIT?

1. JIT compiler compiles instructions into machine code at run time.

2. The code compiler by the JIT compiler runs under CLR.

3. The instructions compiled by JIT compilers are written in native code.

4. The instructions compiled by JIT compilers are written in Intermediate Language (IL) code.

5. The method is JIT compiled even if it is not called

A. 1, 2, 3

B. 2, 4

C. 3, 4, 5

D. 1, 2

113. Which of the following statements are correct about data types?

1. If the integer literal exceeds the range of byte, a compilation error will occur.

2. We cannot implicitly convert non-literal numeric types of larger storage size to byte.

3. Byte cannot be implicitly converted to float.

4. A char can be implicitly converted to only int data type.

5. We can cast the integral character codes.

A. 1, 3, 5

B. 2, 4

C. 3, 5

D. 1, 2, 5

Answer: Option D

114. Which of the following is an 8-byte Integer?

A. Char

B. Long

C. Short

D. Byte

E. Integer

Answer: Option B

115. Which of the following is NOT an Integer?

A. Char

B. Byte

C. Integer

D. Short

E. Long

Answer: Option A

116. Which of the following statements is correct?

A. Information is never lost during narrowing conversions.

B. The CInteger() function can be used to convert a Single to an Integer.

C. Widening conversions take place automatically.

D. Assigning an Integer to an Object type is known as Unboxing.

E. 3.14 can be treated as Decimal by using it in the form 3.14F.

Answer: Option C

117. Which of the following are value types?

1. Integer

2. Array

3. Single

4. String

5. Long

A. 1, 2, 5

B. 1, 3, 5

C. 2, 4

D. 3, 5

Answer: Option B

118. Which of the following does not store a sign?

A. Short

B. Integer

C. Long

D. Byte

E. Single

Answer: Option D

119. What is the size of a Decimal?

A. 4 byte

B. 8 byte

C. 16 byte

D. 32 byte

Answer: Option C

120. Which of the following is the correct size of a Decimaldatatype?

A. 8 Bytes

B. 4 Bytes

C. 10 Bytes

D. 16 Bytes

E. None of the above.

Answer: Option D

121. Which of the following statements are correct?

1. We can assign values of any type to variables of type object.

2. When a variable of a value type is converted to object, it is said to be unboxed.

3. When a variable of type object is converted to a value type, it is said to be boxed.

4. Boolean variable cannot have a value of null.

5. When a value type is boxed, an entirely new object must be allocated and constructed.

A. 2, 5

B. 1, 5

C. 3, 4

D. 2, 3

Answer: Option B

122. Which of the following statements are correct about data types?

1. Each value type has an implicit default constructor that initializes the default value of that type.

2. It is possible for a value type to contain the null value.

3. All value types are derived implicitly from System.ValueType class.

4. It is not essential that local variables in C# must be initialized before being used.

5. Variables of reference types referred to as objects and store references to the actual data.

A. 1, 3, 5

B. 2, 4

C. 3, 5

D. 2, 3, 4

Answer: Option A

123. Which of the following statements are correct about datatypes in C#.NET?

1. Every datatype is either a value type or a reference type.

2. Value types are always created on the heap.

3. Reference types are always created on the stack.

4. Mapping of every value type to a type in Common Type System facilitates Interoperability in C#.NET.

5. Every reference type gets mapped to a type in Common Type System.

A. 1, 3

B. 2, 5

C. 1, 4

D. 3, 4

Answer: Option C

124. Which of the following is the correct default value of a Boolean type?

A. 0 B. 1 C. True D. False E. -1

Answer: Option D

125. Which of the following statements is correct?

A. It is not possible to extend the if statement to handle multiple conditions using the else-if arrangement.

B. The switch statement can include any number of case instances with two case statements having the same value.

C. A jump statement such as a break is required after each case block excluding the last block if it is a default statement.

D. The if statement selects a statement for execution based on the value of a Boolean expression.

E. C# always supports an implicit fall through from one case label to another.

Answer: Option D

126. Which of the following can be used to terminate a while loop and transfer control outside the loop?

1. exit while

2. continue

3. exit statement

4. break

5. goto

A. 1, 3

B. 2, 4

C. 3, 5

D. 4, 5

E. None of these

Answer: Option D

127. Which of the following statements are correct?

1. A switch statement can act on numerical as well as Boolean types.

2. A switch statement can act on characters, strings and enumerations types.

3. We cannot declare variables within a case statement if it is not enclosed by { }.

4. The foreach statement is used to iterate through the collection to get the desired information and should be used to change the contents of the collection to avoid unpredictable side effects.

5. All of the expressions of the for statement are not optional.

A. 1, 2

B. 2, 3

C. 3, 5

D. 4, 5

E. None of these

Answer: Option A

128. Which of the following statements are correct?

1. The switch statement is a control statement that handles multiple selections and enumerations by passing control to one of the case statements within its body.

2. The goto statement passes control to the next iteration of the enclosing iteration statement in which it appears.

3. Branching is performed using jump statements which cause an immediate transfer of the program control.

4. A common use of continue is to transfer control to a specific switch-case label or the default label in a switch statement.

5. The do statement executes a statement or a block of statements enclosed in {}repeatedly until a specified expression evaluates to false.

A. 1, 2, 4

B. 1, 3, 5

C. 2, 3, 4

D. 3, 4, 5

E. None of these

Answer: Option B

129. Which of the following statements is correct?

A. It is not possible to extend the if statement to handle multiple conditions using the else-if arrangement.

B. The switch statement can include any number of case instances with two case statements having the same value.

C. A jump statement such as a break is required after each case block excluding the last block if it is a default statement.

D. The if statement selects a statement for execution based on the value of a Boolean expression.

E. C# always supports an implicit fall through from one case label to another.

Answer: Option D

130. Which of the following can be used to terminate a while loop and transfer control outside the loop?

1. exit while

2. continue

3. exit statement

4. break

5. goto

A. 1, 3

B. 2, 4

C. 3, 5

D. 4, 5

E. None of these

Answer: Option D

131. Which of the following statements are correct?

1. A switch statement can act on numerical as well as Boolean types.

2. A switch statement can act on characters, strings and enumerations types.

3. We cannot declare variables within a case statement if it is not enclosed by { }.

4. The foreach statement is used to iterate through the collection to get the desired information and should be used to change the contents of the collection to avoid unpredictable side effects.

5. All of the expressions of the for statement are not optional.

A. 1, 2

B. 2, 3

C. 3, 5

D. 4, 5

E. None of these

Answer: Option A

132. Which of the following statements are correct?

1. The switch statement is a control statement that handles multiple selections and enumerations by passing control to one of the case statements within its body.

2. The goto statement passes control to the next iteration of the enclosing iteration statement in which it appears.

3. Branching is performed using jump statements which cause an immediate transfer of the program control.

4. A common use of continue is to transfer control to a specific switch-case label or the default label in a switch statement.

5. The do statement executes a statement or a block of statements enclosed in {} repeatedly until a specified expression evaluates to false.

A. 1, 2, 4

B. 1, 3, 5

C. 2, 3, 4

D. 3, 4, 5

E. None of these

Answer: Option B

133. Which of the following is NOT an Arithmetic operator in C#.NET?

A. ** B. + C. / D. % E. *

Answer: Option A

134. Which of the following are NOT Relational operators in C#.NET?

1. >=

2. !=

3. Not

4. <=

5. <>=

A. 1, 3

B. 2, 4

C. 3, 5

D. 4, 5

E. None of these

Answer: Option C

135. Which of the following is NOT a Bitwise operator in C#.NET?

A. & B. | C. << D. ^ E. ~

Answer: Option C

136. Which of the following statements are correct about the Bitwise & operator used in C#.NET?

1. The & operator can be used to Invert a bit.

2. The & operator can be used to put ON a bit.

3. The & operator can be used to put OFF a bit.

4. The & operator can be used to check whether a bit is ON.

5. The & operator can be used to check whether a bit is OFF.

A. 1, 2, 4

B. 2, 3, 5

C. 3, 4

D. 3, 4, 5

E. None of these

Answer: Option D

137. Which of the following are Logical operators in C#.NET?

1. &&

2. ||

3. !

4. Xor

5. %

A. 1, 2, 3

B. 1, 3, 4

C. 2, 4, 5

D. 3, 4, 5

E. None of these

Answer: Option A

138. Which of the following statements is correct about Bitwise | operator used in C#.NET?

A. The | operator can be used to put OFF a bit.

B. The | operator can be used to Invert a bit.

C. The | operator can be used to check whether a bit is ON.

D. The | operator can be used to check whether a bit is OFF.

E. The | operator can be used to put ON a bit.

Answer: Option E

139. Which of the following is NOT an Assignment operator in C#.NET?

A. \=

B. /=

C. *=

D. +=

E. %=

Answer: Option A

140. Which of the following statements is correct about Bitwise ^ operator used in C#.NET?

A. The ^ operator can be used to put ON a bit.

B. The ^ operator can be used to put OFF a bit.

C. The ^ operator can be used to Invert a bit.

D. The ^ operator can be used to check whether a bit is ON.

E. The ^ operator can be used to check whether a bit is OFF.

Answer: Option C

141. Which of the following statements are correct?

1. The conditional operator (?:) returns one of two values depending on the value of a Boolean expression.

2. The as operator in C#.NET is used to perform conversions between compatible reference types.

3. The &* operator is also used to declare pointer types and to dereference pointers.

4. The -> operator combines pointer dereferencing and member access.

5. In addition to being used to specify the order of operations in an expression, brackets [ ] are used to specify casts or type conversions.

A. 1, 2, 4

B. 2, 3, 5

C. 3, 4, 5

D. 1, 3, 5

E. None of these

Answer: Option A

142. Which of the following statements are correct?

1. An argument passed to a ref parameter need not be initialized first.

2. Variables passed as out arguments need to be initialized prior to being passed.

3. Argument that uses params keyword must be the last argument of variable argument list of a method.

4. Pass by reference eliminates the overhead of copying large data items.

5. To use a ref parameter only the calling method must explicitly use the ref keyword.

A. 1, 2

B. 2, 3

C. 3, 4

D. 4, 5

E. None of these

[Dono]

143. A function returns a value, whereas a subroutine cannot return a value.

A. True B. False

Answer: Option A

144. Which of the following statements are correct about functions and subroutines used in C#.NET?

1. A function cannot be called from a subroutine.

2. The ref keyword causes arguments to be passed by reference.

3. While using ref keyword any changes made to the parameter in the method will be reflected in that variable when control passes back to the calling method.

4. A subroutine cannot be called from a function.

5. Functions and subroutines can be called recursively.

A. 1, 2, 4

B. 2, 3, 5

C. 3, 5

D. 4, 5

E. None of these

Answer: Option B

145. Which of the following statements are correct?

1. C# allows a function to have arguments with default values.

2. C# allows a function to have variable number of arguments.

3. Omitting the return value type in method definition results into an exception.

4. Redefining a method parameter in the method's body causes an exception.

5. params is used to specify the syntax for a function with variable number of arguments.

A. 1, 3, 5

B. 3, 4, 5

C. 2, 5

D. 4, 5

E. None of these

Answer: Option C

146. Which of the following statements are correct about functions used in C#.NET?

1. Function definitions cannot be nested.

2. Functions can be called recursively.

3. If we do not return a value from a function then a value -1 gets returned.

4. To return the control from middle of a function exit function should be used.

5. Function calls can be nested.

A. 1, 2, 5

B. 2, 3, 5

C. 2, 3

D. 4, 5

E. None of these

Answer: Option A

147. How many values is a function capable of returning?

A. 1

B. 0

C. Depends upon how many params arguments does it use.

D. Any number of values.

E. Depends upon how many ref arguments does it use.

Answer: Option A

148. How many values is a subroutine capable of returning?

A. Depends upon how many params arguments does it use.

B. Any number of values.

C. Depends upon how many ref arguments does it use.

D. 0

E. 1

Answer: Option D

149. Which of the following CANNOT occur multiple number of times in a program?

A. namespace

B. Entrypoint

C. Class

D. Function

E. Subroutine

Answer: Option B

150. Which of the following statements are correct about subroutines used in C#.NET?

1. If we do not return a value from a subroutine then a value -1 gets returned.

2. Subroutine definitions cannot be nested.

3. Subroutine can be called recursively.

4. To return the control from middle of a subroutine exit subroutine should be used.

5. Subroutine calls can be nested.

A. 1, 2, 3

B. 2, 3, 5

C. 3, 5

D. 3, 4

E. None of these

Answer: Option B

151. A function can be used in an expression, whereas a subroutine cannot be.

A. True B. False

Answer: Option A

152. Which one of the following classes are present System.Collections.Generic namespace?

1. Stack
2. Tree
3. SortedDictionary
4. SortedArray

A. 1 and 2 only

B. 2 and 4 only

C. 1 and 3 only

D. All of the above

E. None of the above

Answer: Option C

153. Which of the following statements are valid about generics in .NET Framework?

1. Generics is a language feature.

2. We can create a generic class, however, we cannot create a generic interface in C#.NET.

3. Generics delegates are not allowed in C#.NET.

4. Generics are useful in collection classes in .NET framework.

5. None of the above

A. 1 and 2 Only

B. 1, 2 and 3 Only

C. 1 and 4 Only

D. All of the above

E. None of the above

Answer: Option C

154. Which of the following statements is valid about generic procedures in C#.NET?

A. All procedures in a Generic class are generic.

B. Only those procedures labeled as Generic are generic.

C. Generic procedures can take at the most one generic parameter.

D. Generic procedures must take at least one type parameter.

E. None of the above.

Answer: Option D

155. Which of the following statements is valid about advantages of generics?

A. Generics shift the burden of type safety to the programmer rather than compiler.

B. Generics require use of explicit type casting.

C. Generics provide type safety without the overhead of multiple implementations.

D. Generics eliminate the possibility of run-time errors.

E. None of the above.

Answer: Option C

1. Which of the following can be declared in an interface?
    1. Properties
    2. Methods
    3. Enumerations
    4. Events
    5. Structures

   A.1, 3
   B.1, 2, 4
   C.3, 5
   D.4, 5

Answer: Option **B**

2. A class implements two interfaces each containing three methods. The class contains no instance data. Which of the following correctly indicate the size of the object created from this class?
   A.12 bytes
   B.24 bytes
   C.0 byte
   D.8 bytes
   E.16 bytes

Answer: Option **B**

3. Which of the following statements is correct about an interface used in C#.NET?

A. One class can implement only one interface.

B. In a program if one class implements an interface then no other class in the same program can implement this interface.

C. From two base interfaces a new interface cannot be inherited.

D. Properties can be declared inside an interface.

E. Interfaces cannot be inherited.

Answer: Option **D**

4. Which of the following statements is correct about Interfaces used in C#.NET?

A. All interfaces are derived from an *Object* class.

B. Interfaces can be inherited.

C. All interfaces are derived from an *Object* interface.

D. Interfaces can contain only method declaration.

E. Interfaces can contain static data and methods.

Answer: Option **B**

5. Which of the following statements is correct about an interface used in C#.NET?

A. If a class implements an interface partially, then it becomes an abstract class.

B. A class cannot implement an interface partially.

C. An interface can contain static methods.

D. An interface can contain static data.

E. Multiple interface inheritance is not allowed.

Answer: Option **A**

6. Which of the following statements is correct about an interface?

A. One interface can be implemented in another interface.

B. An interface can be implemented by multiple classes in the same program.

C. A class that implements an interface can explicitly implement members of that interface.

D. The functions declared in an interface have a body.

Answer: Option **C**

7. Which of the following statements are correct about an interface in C#.NET?

1. A class can implement multiple interfaces.
2. Structures cannot inherit a class but can implement an interface.
3. In C#.NET, *:* is used to signify that a class member implements a specific interface.
4. An interface can implement multiple classes.
5. The static attribute can be used with a method that implements an interface declaration.

A. 1, 2, 3

B. 2, 4

C.3, 5
D.None of the above.

Answer: Option **A**

8. Which of the following statements is correct?

A. When a class inherits an interface it inherits member definitions as well as its implementations.
B. An interface cannot contain the signature of an indexer.
C. Interfaces members are automatically public.
D. To implement an interface member, the corresponding member in the class must be public as well as static.

Answer: Option **C**

9. Which of the following statements are correct about an interface used in C#.NET?

1. An interface can contain properties, methods and events.
2. The keyword must implement forces implementation of an interface.
3. Interfaces can be overloaded.
4. Interfaces can be implemented by a class or a struct.
5. Enhanced implementations of an interface can be developed without breaking existing code.

A.1, 2
B.1, 4, 5
C.3, 4
D.3 only

Answer: Option **B**

10. Which of the following can implement an interface?

1. Data
2. Class
3. Enum
4. Structure
5. Namespace

A.1, 3
B.2, 4
C.3, 5
D.4 only

Answer: Option **B**

11. Which of the following unary operators can be overloaded?

1. *true*
2. *false*
3. +
4. *new*
5. *is*

A.1, 2, 3
B.3, 4, 5
C.3 only
D.5 only

Answer: Option **A**

12. A derived class can stop virtual inheritance by declaring an override as
A.inherits
B.extends
C.inheritable
D.not inheritable
E.sealed

Answer: Option **E**

13. Which of the following keyword is used to change the data and behavior of a base class by replacing a member of a base class with a new derived member?
A.new
B.base
C.overloads
D.override
E.overridable

Answer: Option **A**

14. Which of the following statements is correct?
A.When used as a modifier, the *new* keyword explicitly hides a member inherited from a base class.
B.Operator overloading works in different ways for structures and classes.
C.It is not necessary that all operator overloads are static methods of the class.
D.The cast operator can be overloaded.

Answer: Option **A**

15. Which of the following keyword is used to overload user-defined types by defining static member functions?
A.*op*

B. *opoverload*
C. *operator*
D. *operatoroverload*
E. *udoperator*

Answer: Option **C**

16. Which of the following statements is correct?
A. Static methods can be a virtual method.
B. Abstract methods can be a virtual method.
C. It is necessary to override a virtual method.
D. When overriding a method, the names and type signatures of the override method must be the same as the virtual method that is being overriden.
E. We can override virtual as well as non-virtual methods.

Answer: Option **D**

17.

Which of the followin
delegate?

1. Inheritance is
2. Delegates are
3. Delegates prov
4. The declaratio
signature of th
it.
5. Functions call
bound.

A. 1 and 2 only
B. 1, 2 and 3 only
C. 2, 3 and 4 only
D. All of the above
E. None of the above

Answer: Option **D**

18. Which of the following statements are correct about delegates?

1. Delegates are not type-safe.
2. Delegate is a user-defined type.
3. Only one method can be bound with one delegate object.
4. Delegates can be used to implement callback notification.
5. Delegates permit execution of a method on a secondary thread in an asynchronous manner.

A. 1 and 2 only
B. 1, 2 and 3 only
C. 2, 4 and 5 only
D. 4 and 5 only
E. All of the above

Answer: Option **C**

19. Which of the following statements are correct about delegates?
A. Delegates cannot be used to call a static method of a class.
B. Delegates cannot be used to call procedures that receive variable number of arguments.
C. If signatures of two methods are same they can be called through the same delegate object.
D. Delegates cannot be used to call an instance function. Delegates cannot be used to call an instance subroutine.

Answer: Option **B**

20. Suppose on pushing a button an object is to be notified, but it is not known until runtime which object should be notified. Which of the following programming constructs should be used to implement this idea?
A. Attribute
B. Delegate
C. Namespace
D. Interface
E. Encapsulation

Answer: Option **B**

21. Which of the following statements is incorrect about a delegate?
A. A single delegate can invoke more than one method.
B. Delegates can be shared.
C. Delegate is a value type.
D. Delegates are type-safe wrappers for function pointers.
E. The signature of a delegate must match the signature of the method that is to be called using it.

Answer: Option **C**

22. Suppose a Generic class called *SortObjects* is to be made capable of sorting objects of any type (*Integer*, *Single*, *Byte* etc.). Which of the following programming constructs should be used to implement the comparision function?
A. Namespace
B. Interface
C. Encapsulation
D. Delegate
E. Attribute

Answer: Option **D**

23. With which of the following can the *ref* keyword be used?

1. Static data
2. Instance data
3. Static function/subroutine
4. Instance function/subroutine

A. 1, 2
B. 3, 4
C. 1, 3
D. 2, 4
E. All of the above

Answer: Option **B**

24. Which of the following statements are correct about an *ArrayList* collection that implements the *IEnumerable* interface?

1. The *ArrayList* class contains an inner class that implements the *IEnumerator* interface.
2. An *ArrayList* Collection cannot be accessed simultaneously by different threads.
3. The inner class of *ArrayList* can access *ArrayList* class's members.
4. To access members of *ArrayList* from the inner class, it is necessary to pass *ArrayList* class's reference to it.
5. Enumerator's of *ArrayList* Collection can manipulate the array.

A. 1 and 2 only
B. 1 and 3 and 4 only
C. 2 and 5 only
D. All of the above
E. None of the above

Answer: Option **B**

25. How many enumerators will exist if four threads are simultaneously working on an *ArrayList* object?

A. 1
B. 3
C. 2
D. 4
E. Depends upon the Project Setting made in Visual Studio.NET.

Answer: Option **D**

26. In which of the following collections is the Input/Output index-based?

1. *Stack*
2. *Queue*
3. *BitArray*
4. *ArrayList*
5. *HashTable*

A. 1 and 2 only
B. 3 and 4 only
C. 5 only
D. 1, 2 and 5 only
E. All of the above

Answer: Option **B**

27. Which of the following statements are correct about the *Stack* collection?

1. It can be used for evaluation of expressions.
2. All elements in the *Stack* collection can be accessed using an enumerator.
3. It is used to maintain a FIFO list.
4. All elements stored in a *Stack* collection must be of similar type.
5. Top-most element of the *Stack* collection can be accessed using the *Peek()* method.

A. 1 and 2 only
B. 3 and 4 only
C. 1, 2 and 5 only
D. All of the above
E. None of the above

Answer: Option **C**

28. A *HashTable t* maintains a collection of names of states and capital city of each state. Which of the following is the correct way to find out whether *"Kerala"* state is present in this collection or not?

A. *t.ContainsKey("Kerala");*
B. *t.HasValue("Kerala");*
C. *t.HasKey("Kerala");*
D. *t.ContainsState("Kerala");*
E. *t.ContainsValue("Kerala");*

Answer: Option **A**

29. Which of the following is NOT an interface declared in *System.Collections* namespace?

A. *IComparer*
B. *Enumerable*

C. *Enumerator*
D. *IDictionaryComparer*
E. *IDictionaryEnumerator*

Answer: Option **D**

30. Suppose value of the *Capacity* property of *ArrayList* Collection is set to 4. What will be the capacity of the Collection on adding fifth element to it?
A. 4 B. 8
C. 16 D. 32

31. Which of the following is an ordered collection class?

1. *Map*
2. *Stack*
3. *Queue*
4. *BitArray*
5. *HashTable*

A. 1 only
B. 2 and 3 only
C. 4 and 5 only
D. All of the above
E. None of the above

Answer: Option **B**

32. Which of the following is the correct way to find out the number of elements currently present in an *ArrayList* Collection called *arr*?
A. *arr.Count*
B. *arr.GrowSize*
C. *arr.MaxIndex*
D. *arr.Capacity*
E. *arr.UpperBound*

Answer: Option **A**

33. Which of the following statements are correct about a *HashTable* collection?

1. It is a keyed collection.
2. It is a ordered collection.
3. It is an indexed collection.
4. It implements a *IDictionaryEnumerator* interface in its inner class.
5. The key - value pairs present in a *HashTable* can be accessed using the *Keys* and *Values* properties of the inner class that implements the *IDictionaryEnumerator* interface.

A.1 and 2 only
B.1, 2 and 3 only
C.4 and 5 only
D.1, 4 and 5 only
E. All of the above

Answer: Option **D**

34. Which of the following statements are correct about the *Collection* Classes available in Framework Class Library?
A.Elements of a collection cannot be transmitted over a network.
B.Elements stored in a collection can be retrieved but cannot be modified.
C.It is not easy to adopt the existing Collection classes for newtype of objects.
D.Elements stored in a collection can be modified only if allelements are of similar types.
E. They use efficient algorithms to manage the collection, thereby improving the performance of the program.

Answer: Option **E**

35. Which of the following statements are correct?

1. Instance members of a *class* can be accessed only through an object of that *class*.
2. A *class* can contain only instance data and instance member *function*.
3. All objects created from a *class* will occupy equal number of bytes in memory.
4. A *class* can contain Friend functions.
5. A *class* is a blueprint or a template according to which objects are created.

A.1, 3, 5
B.2, 4
C.3, 5
D.2, 4, 5
E. None of these

Answer: Option **A**

36. Which of the following statements is correct?
A.Procedural Programming paradigm is different than structured programming paradigm.
B. Object Oriented Programming paradigm stresses on dividing the logic into smaller parts and writing procedures for each part.
C.Classes and objects are corner stones of structured programming paradigm.
D. Object Oriented Programming paradigm gives equal importance to data and the procedures that work on the data.
E. C#.NET is a structured programming language.

Answer: Option **D**

37. The *this* reference gets created when a member function (non-shared) of a class is called.
A.True

Answer: Option **A**

38. Which of the following statements are correct?

1. Data membersofa class are by default *public*.
2. Data members of a class are by default *private*.
3. Member functions ofaclass are by default *public*.
4. A private function of a class can access a *public* function within the same class.
5. Member function of a class are by default *private*.

A.1, 3, 5
B.1, 4
C.2, 4, 5
D.1, 2, 3
E. None of these

Answer: Option **C**

39. Which of the following statements are correct about the *this* reference?

1. *this* reference can be modified in the instance member function of a class.
2. Static functions of a class never receive the *this* reference.
3. Instance member functions of a class always receive a *this* reference.
4. *this* reference continues to exist even after control returns from an instance member function.
5. While calling an instance member function we are not required to pass the *this* reference explicitly.

A.1, 4
B.2, 3, 5
C.3, 4
D.2, 5
E. None of these

Answer: Option **B**

40. Which of the following statements are correct about objects of a user-defined class called *Sample*?

1. All objects of *Sample* class will always have exactly same data.
2. Objects of *Sample* class may have same or different data.
3. Whether objects of *Sample* class will have same or different data depends upon a Project Setting made in Visual Studio.NET.

4. Conceptually, each object of *Sample* class will have instance data and instance member functions of the *Sample* class.
5. All objects of *Sample* class will share one copy of member functions.

A. 1, 3
B. 2, 4
C. 4, 5
D. 3, 5
E. None of these

Answer: Option **C**

41. Which of the following statements is correct about classes and objects in C#.NET?
A. Class is a value type.
B. Since objects are typically big in size, they are created on the stack.
C. Objects of smaller size are created on the heap.
D. Smaller objects that get created on the stack can be given names.
E. Objects are always nameless.

Answer: Option **E**

42. The [Serializable()] attribute gets inspected at
A. Compile-time
B. Run-time
C. Design-time
D. Linking-time
E. None of the above

Answer: Option **B**

43. Which of the following are correct ways to specify the targets for a custom attribute?
A. By applying *AttributeUsage* to the custom attribute's class definition.
B. By applying *UsageAttribute* to the custom attribute's class definition.
C. Once an attribute is declared it applies to all the targets.
D. By applying *AttributeUsageAttribute* to the custom attribute's class definition.
E. None of the above.

Answer: Option **D**

44. Which of the following are correct ways to pass a parameter to an attribute?

1. By value
2. By reference
3. By address

4. By position
5. By name

A.1, 2
B.1, 2, 3
C.4, 5
D.All of the above

Answer: Option **C**

45. Which of the following statements are correct about inspecting an attribute in C#.NET?

1. An attribute can be inspected at link-time.
2. An attribute can be inspected at compile-time.
3. An attribute can be inspected at run-time.
4. An attribute can be inspected at design-time.

A.1, 2
B.3, 4
C.1, 3, 4
D.All of the above
E. None of the above

Answer: Option **A**

46. Which of the following statements are correct about Attributes used in C#.NET?

A. If there is a custom attribute *BugFixAttribute* then the compiler will look ONLY for the *BugFix* attribute in the code that uses this attribute.
B. To create a custom attribute we need to create a custom attribute structure and derive it from *System.Attribute*.
C. To create a custom attribute we need to create a class and implement *IAttribute* interface in it.
D. If a *BugFixAttribute* is to receive three parameters then the *BugFixAttribute* class should implement a zero-argument constructor.
E. The CLR can change the behaviour of the code depending upon the attributes applied to it.

Answer: Option **E**

47. Which of the following statements are correct about Attributes in C#.NET?

1. On compiling a C#.NET program the attibutes applied are recorded in the metadata of the assembly.
2. On compilation all the attribute's tags are deleted from the program.
3. It is not possible to create custom attributes..
4. The attributes applied can be read from an assembly using Reflection class.

5. An attribute can have parameters.

A.1 and 2 only
B.2 and 4 only
C.1, 4 and 5 only
D.All of the above
E. None of the above

Answer: Option **C**

48. Which of the following correctly describes the contents of the filename AssemblyInfo.cs?
A.It contains method-level attributes.
B.It contains class-level attributes.
C.It contains assembly-level attributes.
D.It contains structure-level attributes.
E. It contains namespace-level attributes.

Answer: Option **C**

49. It possible to create a custom attribute that can be applied only to specific programming element(s) like ____ .
A.Classes
B.Methods
C.Classes and Methods
D.Classes, Methods and Member-Variables

Answer: Option **C**

50. Which of the following CANNOT be a target for a custom attribute?
A.Enum
B.Event
C.Delegate
D.Interface
E. Namespace

Answer: Option **E**

51. Once applied which of the following CANNOT inspect the applied attribute?
A.CLR
B.Linker
C.ASP.NET Runtime
D.Visual Studio.NET
E. Language compilers

Answer: Option **B**

52. Attributes can be applied to

1. Method
2. Class
3. Assembly
4. Namespace
5. Enum

A. 1 and 2 only
B. 1, 2 and 3
C. 4 and 5 only
D. All of the above
E. None of the above

Answer: Option **B**.

53. If a namespace is present in a library then which of the following is the correct way to use the elements of the namespace?

A. Add Reference of the namespace.
Use the elements of the namespace.
B. Add Reference of the namespace.
Import the namespace.
Use the elements of the namespace.
C. Import the namespace.
Use the elements of the namespace.
D. Copy the library in the same directory as the project that is trying to use it.
Use the elements of the namespace.
E. Install the namespace in Global Assembly Cache.
Use the elements of the namespace.

Answer: Option **B**

54. Which of the following is NOT a namespace in the .NET Framework Class Library?

A. *System.Process*
B. *System.Security*
C. *System.Threading*
D. *System.Drawing*
E. *System.Xml*

Answer: Option **A**

55. Which of the following statements is correct about a namespace in C#.NET?
A. Namespaces help us to control the visibility of the elements present in it.
B. A namespace can contain a class but not another namespace.
C. If not mentioned, then the name 'root' gets assigned to the namespace.
D. It is necessary to use the using statement to be able to use an element of a namespace.

E. We need to organise the classes declared in Framework Class Library into different namespaces.

Answer: Option **A**

56. Which of the following is absolutely neccessary to use a class *Point* present in namespace *Graph* stored in library?
A. Use fully qualified name of the *Point* class.
B. Use using statement before using the *Point* class.
C. Add Reference of the library before using the *Point* class.
D. Use using statement before using the *Point* class.
E. Copy the library into the current project directory before using the *Point* class.

Answer: Option **C**

57. Which of the followings are NOT a .NET namespace?

1. *System.Web*
2. *System.Process*
3. *System.Data*
4. *System.Drawing2D*
5. *System.Drawing3D*

A. 1, 3
B. 2, 4, 5
C. 3, 5
D. 1, 2, 3

Answer: Option **B**

58. Which of the following statements is correct about namespaces in C#.NET?
A. Namespaces can be nested only up to level 5.
B. A namespace cannot be nested.
C. There is no limit on the number of levels while nesting namespaces.
D. If namespaces are nested, then it is necessary to use using statement while using the elements of the inner namespace.
E. Nesting of namespaces is permitted, provided all the inner namespaces are declared in the same file.

Answer: Option **C**

59. Which of the following statements is correct about the using statement used in C#.NET?
A. *using* statement can be placed anywhere in the C#.NET source code file.
B. It is permitted to define a member at namespace level as a *using* alias.
C. A C#.NET source code file can contain any number of *using* statement.
D. By using *using* statement it is possible to create an alias for the namespace but not for the

namespace element.

E. By using *using* statement it is possible to create an alias for the namespace element but not for the namespace.

Answer: Option **C**

60. Which of the following statements are correct about a namespace used in C#.NET?

1. Classes must belong to a namespace, whereas structures need not.
2. Every *class*, struct, *enum*, delegate and interlace has to belong to some or the other namespace.
3. All elements of the namespace have to belong to one file.
4. If not mentioned, a namespace takes the name of the current project.
5. The namespace should be imported to be able to use the elements in it.

A.1, 3
B.2, 4, 5
C.3, 5
D.4 only

Answer: Option **B**

61. Which of the following CANNOT belong to a C#.NET Namespace?

A.Class B.struct
C.Enum D.Data
E. Interface

Answer: Option **D**

62. Which of the following statements is correct about a namespace used in C#.NET?
A.Nested namespaces are not allowed.
B.Importing outer namespace imports inner namespace.
C.Nested namespaces are allowed.
D.If nested, the namespaces cannot be split across files.

Answer: Option **C**

63. A property can be declared inside a class, struct, Interface.
A.True B.False

Answer: Option **A**

64. Which of the following statements is correct about properties used in C#.NET?
A.A property can simultaneously be read only or write only.
B.A property can be either read only or write only.
C.A write only property will have only get accessor.

D.A write only property will always return a value.

Answer: Option **B**

65. A *Student* class has a property called *rollNo* and *stu* is a reference to a *Student* object and we want the statement *stu.RollNo = 28* to fail. Which of the following options will ensure this functionality?
A.Declare *rollNo* property with both get and set accessors.
B.Declare *rollNo* property with only set accessor.
C.Declare *rollNo* property with get, set and normal accessors.
D.Declare *rollNo* property with only get accessor.
E.None of the above

Answer: Option **D**

66. Which of the following statements are correct?

1. The signature of an indexer consists of the number and types of its formal parameters.
2. Indexers are similar to properties except that their accessors take parameters.
3. Accessors of interface indexers use modifiers.
4. The type of an indexer and the type of its parameters must be at least as accessible as the indexer itself.
5. An interface accessor contains a body.

A.1, 3, 5
B.1, 2, 4
C.3, 5
D.2, 4

Answer: Option **B**

67. A property can be declared inside a namespace or a procedure.
A.True B.False

Answer: Option **B**

68.Which of the following statements is correct about properties used in C#.NET?

A.Every property must have a set accessor and a get accessor.
B.Properties cannot be overloaded.
C.Properties of a class are actually methods that work like data members.
D.A property has to be either read only or a write only.

Answer: Option **C**

69. Which of the folowing does an indexer allow to index in the same way as an array?

1. A class
2. A property
3. A struct
4. A function
5. An interface

A.1, 3, 5
B.2, 4
C.3, 5
D.3, 4, 5

Answer: Option **A**

70. An *Employee* class has a property called *age* and *emp* is reference to a *Employee* object and we want the statement *Console.WriteLine(emp.age)* to fail. Which of the following options will ensure this functionality?
A.Declare *age* property with only get accessor.
B.Declare *age* property with only set accessor.
C.Declare *age* property with both get and set accessors.
D.Declare *age* property with get, set and normal accessors.
E.None of the above

Answer: Option **B**

71. Which of the following statements are correct about an *enum* used inC#.NET?

1. By default the first enumerator has the value equal to the number of elements present in the list.
2. The value of each successive enumerator is decreased by 1.
3. An enumerator contains white space in its name.
4. A variable cannot be assigned to an *enum* element.
5. Values of *enum* elements cannot be populated from a database.

A.1, 2
B.3, 4
C.4, 5
D.1, 4

Answer: Option **C**

72. Which of the following statements is true about an *enum* used in C#.NET?

A.An implicit cast is needed to convert from *enum* type to an integral type.
B.An *enum* variable cannot have a *public* access modifier.
C.An *enum* variable cannot have a *private* access modifier.
D.An *enum* variable can be defined inside a class or a namespace.

E. An *enum* variable cannot have a *protected* access modifier.

Answer: Option **D**

73. Which of the following statements are correct about an *enum* used inC#.NET?

1. To use the keyword *enum*, we should either use [*enum*] or *System.Enum*.
2. *enum* is a keyword.
3. Enum is class declared in *System.Type* namespace.
4. Enum is a class declared in the current project's root namespace.
5. Enum is a class declared in *System* namespace.

A.1, 3
B.2, 4
C.2, 5
D.3, 4

Answer: Option **C**

74. An *enum* that is declared inside a class, struct, namespace or interface is treated as public.
A.True B.False

Answer: Option **A**

75. An *enum* can be declared inside a class, struct, namespace or interface.
A.True B.False

Answer: Option **A**

76. Which of the following CANNOT be used as an underlying datatype for an *enum* in C#.NET?
A.*Byte*
B.*Short*
C.*Float*
D.*Int*

Answer: Option **C**

77.Which of the following statements are correct about *enum* used in C#.NET?

1. Every *enum* is derived from an Object class.
2. Every *enum* is a value type.
3. There does not exist a way to print an element of an *enum* as a string.
4. Every *enum* is a reference type.
5. The *default* underlying datatype of an *enum* is *int*.

A.1, 2, 5
B.1, 4
C.3, 5
D.2, 3, 4

Answer: Option **A**

78. Which of the following statements is correct about an *enum* used in C#.NET?
A.*enum* is a reference type.
B.*enum* is a value type.
C.Whether it a value type or a reference type depends upon size.
D.Whether it a value type or a reference type depends upon a Project Setting made in Visual Stiiclio.NET.
E.We can programmatically control whether it is a value type or a reference type.

Answer: Option **B**

79. Which of the following statements are correct about an *enum* used in C#.NET?

1. An *enum* can be declared inside a class.
2. An *enum* can take Single, Double or Decimal values.
3. An *enum* can be declared outside a class.
4. An *enum* can be declared inside/outside a namespace.
5. An object can be assigned to an *enum* variable.

A.1, 3, 4
B.2, 5
C.3, 4
D.2, 4, 5

Answer: Option **A**

80. The string built using the String class are immutable (unchangeable), whereas, the ones built- using the StringBuilder class are mutable.
A.True B.False

Answer: Option **A**

11.

Which of the following statements about a String is correct?

A.A String is created on the stack.
B.Whether a String is created on the stack or the heap depends on the length of the String.
C.A String is a primitive.
D.A String can be created by using the statement *String s1 = new String;*

E. A String is created on the heap.

Answer: Option **E**

81. Which of the following statement is correct about a String in C#.NET?
A. A String is mutable because it can be modified once it has been created.
B. Methods of the *String* class can be used to modify the string.
C. A number CANNOT be represented in the form of a String.
D. A String has a zero-based index.
E. The *System.Array* class is used to represent a string.

Answer: Option **D**

82. Which of the following statements are correct about the String Class in C#.NET?

1. Two strings can be concatenated by using an expression of the form *s3 = s1 + s2;*
2. String is a primitive in C#.NET.
3. A string built using *StringBuilder* Class is Mutable.
4. A string built using *String* Class is Immutable.
5. Two strings can be concatenated by using an expression of the form *s3 = s1&s2;*

A. 1, 2, 5
B. 2, 4
C. 1, 3, 4
D. 3, 5

Answer: Option **C**

83. Which of the following statements are correct?

1. String is a value type.
2. String literals can contain any character literal including escape sequences.
3. The equality operators are defined to compare the values of string objects as well as references.
4. Attempting to access a character that is outside the bounds of the string results in an *IndexOutOfRangeException*.
5. The contents of a string object can be changed after the object is created.

A. 1, 3
B. 3, 5
C. 2, 4
D. 1, 2, 4

Answer: Option **C**

84. Which one of the following statements is correct?

A.Array elements can be of integer type only.
B.The rank of an Array is the total number of elements it can contain.
C.The length of an Array is the number of dimensions in the Array.
D.The default value of numeric array elements is zero.
E.The Array elements are guaranteed to be sorted.

Answer: Option **D**

85. Which of the following statements are correct about arrays used in C#.NET?

1. Arrays can be rectangular or jagged.
2. Rectangular arrays have similar rows stored in adjacent memory locations.
3. Jagged arrays do not have an access to the methods of *System.Array* Class.
4. Rectangular arrays do not have an access to the methods of *System.Array* Class.
5. Jagged arrays have dissimilar rows stored in non-adjacent memory locations.

A.1, 2
B.1, 3, 5
C.3, 4
D.1, 2, 5
E.4, 5

Answer: Option **D**

86. The space required for structure variables is allocated on stack.
A.True B.False

Answer: Option **A**

87. Creating empty structures is allowed in C#.NET.
A.True B.False

Answer: Option **B**

88. Which of the following statements are correct?

1. A struct can contain properties.
2. A struct can contain constructors.
3. A struct can contain protected data members.
4. A struct cannot contain methods.
5. A struct cannot contain constants.

A.1, 2
B.3, 4
C.1, 2, 4
D.3, 5

Answer: Option **A**

89. C#.NET structures are always value types.
A.True B.False

Answer: Option **A**

90. When would a structure variable get destroyed?
A.When no reference refers to it, it will get garbage collected.
B.Depends upon whether it is created using *new* or without using *new*.
C.When it goes out of scope.
D.Depends upon the Project Settings made in Visual Studio.NET.
E.Depends upon whether we free it's memory using *free()* or *delete()*.

Answer: Option **C**

91. Which of the following statements is correct?
A.A struct never declares a *default* constructor.
B.All value types in C# inherently derive from *ValueType*, which inherits from *Object*.
C.A struct never declares a *default* destructor.
D.In C#, classes and structs are semantically same.

Answer: Option **B**

92. Which of the following are true about classes and struct?

1. A *class* is a reference type, whereas a *struct* is a value type.
2. Objects are created using *new*, whereas structure variables can be created either using *new* or without using *new*.
3. A structure variable will always be created slower than an object.
4. A structure variable will die when it goes out of scope.
5. An object will die when it goes out of scope.

A.1, 2, 4
B.3, 5
C.2, 4
D.3, 4, 5

Answer: Option **A**

93. Which of the following statements are correct about Structures used in C#.NET?

1. A Structure can be declared within a procedure.
2. Structs can implement an *interface* but they cannot inherit from another *struct*.
3. struct members cannot be declared as *protected*.
4. A Structure can be empty.

5. It is an error to initialize an instance field in a *struct*.

A.1, 2, 4
B.2, 3, 5
C.2, 4
D.1, 3

Answer: Option **B**

94. Which of the following statements are TRUE about the .NET CLR?

1. It provides a language-neutral development & execution environment.
2. It ensures that an application would not be able to access memory that it is not authorized to access.
3. It provides services to run "managed" applications.
4. The resources are garbage collected.
5. It provides services to run "unmanaged" applications.

A.Only 1 and 2
B.Only 1, 2 and 4
C.1, 2, 3, 4
D.Only 4 and 5
E. Only 3 and 4

Answer: Option **C**

95. Which of the following are valid .NET CLR JIT performance counters?

1. Total memory used for JIT compilation
2. Average memory used for JIT compilation
3. Number of methods that failed to compile with the standard JIT
4. Percentage of processor time spent performing JIT compilation
5. Percentage of memory currently dedicated for JIT compilation

A.1, 5
B.3, 4
C.1, 2
D.4, 5

Answer: Option **B**

96. Which of the following statements is correct about Managed Code?
A.Managed code is the code that is compiled by the JIT compilers.
B.Managed code is the code where resources are Garbage Collected.
C.Managed code is the code that runs on top of Windows.

D.Managed code is the code that is written to target the services of the CLR.
E. Managed code is the code that can run on top of Linux.

Answer: Option **D**

97. Which of the following utilities can be used to compile managed assemblies into processor-specific native code?

A.gacutil
B.ngen
C.sn
D.dumpbin
E. ildasm

Answer: Option **B**

98. Which of the following are NOT true about .NET Framework?

1. It provides a consistent object-oriented programming environment whether object code is stored and executed locally, executed locally but Internet-distributed, or executed remotely.
2. It provides a code-execution environment that minimizes software deployment and versioning conflicts.
3. It provides a code-execution environment that promotes safe execution of code, including code created by an unknown or semi-trusted third party.
4. It provides different programming models for Windows-based applications and Web-based applications.
5. It provides an event driven programming model for building Windows Device Drivers.

A.1, 2
B.2, 4
C.4, 5
D.1, 2, 4

Answer: Option **C**

99. Which of the following components of the .NET framework provide an extensible set of classes that can be used by any .NET compliant programming language?

A..NET class libraries
B.Common Language Runtime
C.Common Language Infrastructure
D.Component Object Model
E. Common Type System

Answer: Option **A**

100. Which of the following jobs are NOT performed by Garbage Collector?

1. Freeing memory on the stack.
2. Avoiding memory leaks.
3. Freeing memory occupied by unreferenced objects.
4. Closing unclosed database collections.
5. Closing unclosed files.

A.1, 2, 3
B.3, 5
C.1, 4, 5
D.3, 4

Answer: Option **C**

101. Which of the following .NET components can be used to remove unused references from the managed heap?
A.Common Language Infrastructure
B.CLR
C.Garbage Collector
D.Class Loader
E. CTS

Answer: Option **C**

102. Which of the following statements correctly define .NET Framework?
A. It is an environment for developing, building, deploying and executing Desktop Applications, Web Applications and Web Services.
B. It is an environment for developing, building, deploying and executing only Web Applications.
C. It is an environment for developing, building, deploying and executing Distributed Applications.
D.It is an environment for developing, building, deploying and executing Web Services.
E. It is an environment for development and execution of Windows applications.

Answer: Option **A**

103. Which of the following constitutes the .NET Framework?

1. ASP.NET Applications
2. CLR
3. Framework Class Library
4. WinForm Applications
5. Windows Services

A.1, 2
B.2, 3

C.3, 4
D.2, 5

Answer: Option **B**

104. Which of the following assemblies can be stored in Global Assembly Cache?
A.Private Assemblies
B.Friend Assemblies
C.Shared Assemblies
D.Public Assemblies
E. Protected Assemblies

Answer: Option **C**

105. Code that targets the Common Language Runtime is known as
A.Unmanaged
B.Distributed
C.Legacy
D.Managed Code
E. Native Code

Answer: Option **D**

106. Which of the following statements is correct about the .NET Framework?
A..NET Framework uses DCOM for achieving language interoperability.
B..NET Framework is built on the DCOM technology.
C. .NET Framework uses DCOM for making transition between managed and unmanaged code.
D..NET Framework uses DCOM for creating unmanaged applications.
E. .NET Framework uses COM+ services while creating Distributed Applications.

Answer: Option **C**

107. Which of the following is the root of the .NET type hierarchy?
A.*System.Object*
B.*System.Type*
C.*System.Base*
D.*System.Parent*
E. *System.Root*

Answer: Option **A**

108. Which of the following benefits do we get on running managed code under CLR?

1. Type safety of the code running under CLR is assured.
2. It is ensured that an application would not access the memory that it is not

authorized to access.
3. It launches separate process for every application running under it.
4. The resources are Garbage collected.

A.Only 1 and 2
B.Only 2, 3 and 4
C.Only 1, 2 and 4
D.Only 4
E. All of the above

Answer: Option **E**

109. Which of the following security features can .NET applications avail?

1. PIN Security
2. Code Access Security
3. Role Based Security
4. Authentication Security
5. Biorhythm Security

A.1, 4, 5
B.2, 5
C.2, 3
D.3, 4

Answer: Option **C**

110. Which of the following jobs are done by Common Language Runtime?

1. It provides core services such as memory management, thread management, and remoting.
2. It enforces strict type safety.
3. It provides Code Access Security.
4. It provides Garbage Collection Services.

A.Only 1 and 2
B.Only 3, 4
C.Only 1, 3 and 4
D.Only 2, 3 and 4
E. All of the above

Answer: Option **E**

111. Which of the following statements are correct about a .NET Assembly?

1. It is the smallest deployable unit.

2. Each assembly has only one entry point - *Main()*, *WinMain()* or *DLLMain()*.
3. An assembly can be a Shared assembly or a Private assembly.
4. An assembly can contain only code and data.
5. An assembly is always in the form of an EXE file.

A.1, 2, 3
B.2, 4, 5
C.1, 3, 5
D.1, 2

Answer: Option **A**

112. Which of the following statements are correct about JIT?

1. JIT compiler compiles instructions into machine code at run time.
2. The code compiler by the JIT compiler runs under CLR.
3. The instructions compiled by JIT compilers are written in native code.
4. The instructions compiled by JIT compilers are written in Intermediate Language (IL) code.
5. The method is JIT compiled even if it is not called

A.1, 2, 3
B.2, 4
C.3, 4, 5
D.1, 2
20.
Which of the following are parts of the .NET Framework?

1. The Common Language Runtime (CLR)
2. The Framework Class Libraries (FCL)
3. Microsoft Published Web Services
4. Applications deployed on IIS
5. Mobile Applications

A.Only 1, 2, 3
B.Only 1, 2
C.Only 1, 2, 4
D.Only 4, 5
E. All of the above

Answer: Option **B**

113. Which of the following statements are correct about data types?

1. If the integer literal exceeds the range of byte, a compilation error will occur.
2. We cannot implicitly convert non-literal numeric types of larger storage size to byte.
3. Byte cannot be implicitly converted to float.
4. A char can be implicitly converted to only int data type.
5. We can cast the integral character codes.

A.1, 3, 5
B.2, 4
C.3, 5
D.1, 2, 5

Answer: Option **D**

114. Which of the following is an 8-byte Integer?
A.Char
B.Long
C.Short
D.Byte
E. Integer

Answer: Option **B**

115. Which of the following is NOT an Integer?
A.Char
B.Byte
C.Integer
D.Short
E. Long

Answer: Option **A**

116. Which of the following statements is correct?
A.Information is never lost during narrowing conversions.
B.The *CInteger()* function can be used to convert a Single to an Integer.
C.Widening conversions take place automatically.
D.Assigning an Integer to an Object type is known as Unboxing.
E. 3.14 can be treated as Decimal by using it in the form *3.14F*.

Answer: Option **C**

117. Which of the following are value types?

1. *Integer*
2. *Array*
3. *Single*
4. *String*

5. *Long*

A.1, 2, 5
B.1, 3, 5
C.2, 4
D.3, 5

Answer: Option **B**

118. Which of the following does not store a sign?
A.*Short*
B.*Integer*
C.*Long*
D.*Byte*
E.*Single*

Answer: Option **D**

119. What is the size of a Decimal?
A.4 byte
B.8 byte
C.16 byte
D.32 byte

Answer: Option **C**

120. Which of the following is the correct size of a *Decimal*datatype?
A.8 Bytes
B.4 Bytes
C.10 Bytes
D.16 Bytes
E. None of the above.

Answer: Option **D**

121. Which of the following statements are correct?

1. We can assign values of any type to variables of type object.
2. When a variable of a value type is converted to object, it is said to be unboxed.
3. When a variable of type object is converted to a value type, it is said to be boxed.
4. Boolean variable cannot have a value of *null*.
5. When a value type is boxed, an entirely new object must be allocated and constructed.

A.2, 5
B.1, 5

C.3, 4
D.2, 3

Answer: Option **B**

122. Which of the following statements are correct about data types?

1. Each value type has an implicit default constructor that initializes the default value of that type.
2. It is possible for a value type to contain the *null* value.
3. All value types are derived implicitly from *System.ValueType* class.
4. It is not essential that local variables in C# must be initialized before being used.
5. Variables of reference types referred to as objects and store references to the actual data.

A.1, 3, 5
B.2, 4
C.3, 5
D.2, 3, 4

Answer: Option **A**

123. Which of the following statements are correct about datatypes in C#.NET?

1. Every datatype is either a value type or a reference type.
2. Value types are always created on the heap.
3. Reference types are always created on the stack.
4. Mapping of every value type to a type in Common Type System facilitates Interoperability in C#.NET.
5. Every reference type gets mapped to a type in Common Type System.

A.1, 3
B.2, 5
C.1, 4
D.3, 4

Answer: Option **C**

124. Which of the following is the correct default value of a *Boolean* type?

A.0
B.1
C.True
D.False
E. -1

Answer: Option **D**

125. Which of the following statements is correct?

A. It is not possible to extend the *if* statement to handle multiple conditions using the *else-if* arrangement.
B. The *switch* statement can include any number of *case* instances with two *case* statements having the same value.
C. A jump statement such as a *break* is required after each *case* block excluding the last block *if* it is a *default* statement.
D. The *if* statement selects a statement for execution based on the value of a Boolean expression.
E. C# always supports an implicit fall through from one *case* label to another.

Answer: Option **D**

126. Which of the following can be used to terminate a *while* loop and transfer control outside the loop?

1. exit while
2. continue
3. exit statement
4. break
5. goto

A. 1, 3
B. 2, 4
C. 3, 5
D. 4, 5
E. None of these

Answer: Option **D**

127. Which of the following statements are correct?

1. A *switch* statement can act on numerical as well as *Boolean* types.
2. A *switch* statement can act on *characters*, *strings* and *enumerations* types.
3. We cannot declare variables within a *case* statement *if* it is not enclosed by { }.
4. The *foreach* statement is used to iterate through the collection to get the desired information and should be used to change the contents of the collection to avoid unpredictable side effects.
5. All of the expressions of the *for* statement are not optional.

A. 1, 2
B. 2, 3
C. 3, 5
D. 4, 5
E. None of these

Answer: Option **A**

128. Which of the following statements are correct?

1. The *switch* statement is a control statement that handles multiple selections and enumerations by passing control to one of the *case* statements within its body.
2. The *goto* statement passes control to the next iteration of the enclosing iteration statement in which it appears.
3. Branching is performed using jump statements which cause an immediate transfer of the program control.
4. A common use of *continue* is to transfer control to a specific *switch-case* label or the *default* label in a *switch* statement.
5. The *do* statement executes a statement or a block of statements enclosed in *{}*repeatedly until a specified expression evaluates to *false*.

A. 1, 2, 4
B. 1, 3, 5
C. 2, 3, 4
D. 3, 4, 5
E. None of these

Answer: Option **B**

129. Which of the following statements is correct?

A. It is not possible to extend the *if* statement to handle multiple conditions using the *else-if* arrangement.
B. The *switch* statement can include any number of *case* instances with two *case* statements having the same value.
C. A jump statement such as a *break* is required after each *case* block excluding the last block *if* it is a *default* statement.
D. The *if* statement selects a statement for execution based on the value of a Boolean expression.
E. C# always supports an implicit fall through from one *case* label to another.

Answer: Option **D**

130. Which of the following can be used to terminate a *while* loop and transfer control outside the loop?

1. exit while
2. continue
3. exit statement
4. break
5. goto

A. 1, 3
B. 2, 4
C. 3, 5

D.4, 5
E. None of these

Answer: Option **D**

131. Which of the following statements are correct?

1. A *switch* statement can act on numerical as well as *Boolean* types.
2. A *switch* statement can act on *characters*, *strings* and *enumerations* types.
3. We cannot declare variables within a *case* statement *if* it is not enclosed by *{ }*.
4. The *foreach* statement is used to iterate through the collection to get the desired information and should be used to change the contents of the collection to avoid unpredictable side effects.
5. All of the expressions of the *for* statement are not optional.

A.1, 2
B.2, 3
C.3, 5
D.4, 5
E. None of these

Answer: Option **A**

132. Which of the following statements are correct?

1. The *switch* statement is a control statement that handles multiple selections and enumerations by passing control to one of the *case* statements within its body.
2. The *goto* statement passes control to the next iteration of the enclosing iteration statement in which it appears.
3. Branching is performed using jump statements which cause an immediate transfer of the program control.
4. A common use of *continue* is to transfer control to a specific *switch-case* label or the *default* label in a *switch* statement.
5. The *do* statement executes a statement or a block of statements enclosed in *{}* repeatedly until a specified expression evaluates to *false*.

A.1, 2, 4
B.1, 3, 5
C.2, 3, 4
D.3, 4, 5
E. None of these

Answer: Option **B**

133. Which of the following is NOT an Arithmetic operator in C#.NET?

A.** B.+

C./ D.%
E. *

Answer: Option **A**

134. Which of the following are NOT Relational operators in C#.NET?

1. >=
2. !=
3. Not
4. <=
5. <>=

A.1, 3
B.2, 4
C.3, 5
D.4, 5
E. None of these

Answer: Option **C**

135. Which of the following is NOT a Bitwise operator in C#.NET?

A.& B.|
C.<< D.^
E. ~

Answer: Option **C**

136. Which of the following statements are correct about the Bitwise & operator used in C#.NET?

1. The & operator can be used to Invert a bit.
2. The & operator can be used to put ON a bit.
3. The & operator can be used to put OFF a bit.
4. The & operator can be used to check whether a bit is ON.
5. The & operator can be used to check whether a bit is OFF.

A.1, 2, 4
B.2, 3, 5
C.3, 4
D.3, 4, 5
E. None of these

Answer: Option **D**

137. Which of the following are Logical operators in C#.NET?

1. &&
2. ||
3. !
4. Xor
5. %

A.1, 2, 3
B.1, 3, 4
C.2, 4, 5
D.3, 4, 5
E. None of these

Answer: Option **A**

138. Which of the following statements is correct about Bitwise | operator used in C#.NET?

A.The | operator can be used to put OFF a bit.
B.The | operator can be used to Invert a bit.
C.The | operator can be used to check whether a bit is ON.
D.The | operator can be used to check whether a bit is OFF.
E. The | operator can be used to put ON a bit.

Answer: Option **E**

139. Which of the following is NOT an Assignment operator in C#.NET?

A.\=
B./=
C.*=
D.+=
E. %=

Answer: Option **A**

140. Which of the following statements is correct about Bitwise ^ operator used in C#.NET?

A.The ^ operator can be used to put ON a bit.
B.The ^ operator can be used to put OFF a bit.
C. The ^ operator can be used to Invert a bit.
D.The ^ operator can be used to check whether a bit is ON.
E. The ^ operator can be used to check whether a bit is OFF.

Answer: Option **C**

141.Which of the following statements are correct?

1. The conditional operator *(?:)* returns one of two values depending on the value of a Boolean expression.
2. The *as* operator in C#.NET is used to perform conversions between compatible

reference types.
3. The &* operator is also used to declare pointer types and to dereference pointers.
4. The -> operator combines pointer dereferencing and member access.
5. In addition to being used to specify the order of operations in an expression, brackets *[ ]* are used to specify casts or type conversions.

A.1, 2, 4
B.2, 3, 5
C.3, 4, 5
D.1, 3, 5
E. None of these

Answer: Option **A**

142. Which of the following statements are correct?

1. An argument passed to a ref parameter need not be initialized first.
2. Variables passed as out arguments need to be initialized prior to being passed.
3. Argument that uses params keyword must be the last argument of variable argument list of a method.
4. Pass by reference eliminates the overhead of copying large data items.
5. To use a ref parameter only the calling method must explicitly use the *ref* keyword.

A.1, 2
B.2, 3
C.3, 4
D.4, 5
E. None of these

143. A function returns a value, whereas a subroutine cannot return a value.
A.True B.False

Answer: Option **A**

144. Which of the following statements are correct about functions and subroutines used in C#.NET?

1. A function cannot be called from a subroutine.
2. The *ref* keyword causes arguments to be passed by reference.
3. While using ref keyword any changes made to the parameter in the method will be reflected in that variable when control passes back to the calling method.
4. A subroutine cannot be called from a function.
5. Functions and subroutines can be called recursively.

A.1, 2, 4
B.2, 3, 5
C.3, 5

D.4, 5
E. None of these

Answer: Option **B**

145. Which of the following statements are correct?

1. C# allows a function to have arguments with default values.
2. C# allows a function to have variable number of arguments.
3. Omitting the return value type in method definition results into an exception.
4. Redefining a method parameter in the method's body causes an exception.
5. params is used to specify the syntax for a function with variable number of arguments.

A.1, 3, 5
B.3, 4, 5
C.2, 5
D.4, 5
E. None of these

Answer: Option **C**

146. Which of the following statements are correct about functions used in C#.NET?

1. Function definitions cannot be nested.
2. Functions can be called recursively.
3. If we do not return a value from a function then a value *-1* gets returned.
4. To return the control from middle of a function exit function should be used.
5. Function calls can be nested.

A.1, 2, 5
B.2, 3, 5
C.2, 3
D.4, 5
E. None of these

Answer: Option **A**

147. How many values is a function capable of returning?
A.1
B.0
C.Depends upon how many params arguments does it use.
D.Any number of values.
E. Depends upon how many ref arguments does it use.

Answer: Option **A**

148. How many values is a subroutine capable of returning?

A. Depends upon how many params arguments does it use.
B. Any number of values.
C. Depends upon how many ref arguments does it use.
D. 0
E. 1

Answer: Option **D**

149. Which of the following CANNOT occur multiple number of times in a program?

A. Namespace
B. Entrypoint
C. Class
D. Function
E. Subroutine

Answer: Option **B**

150. Which of the following statements are correct about subroutines used in C#.NET?

1. If we do not return a value from a subroutine then a value *-1* gets returned.
2. Subroutine definitions cannot be nested.
3. Subroutine can be called recursively.
4. To return the control from middle of a subroutine exit subroutine should be used.
5. Subroutine calls can be nested.

A. 1, 2, 3
B. 2, 3, 5
C. 3, 5
D. 3, 4
E. None of these

Answer: Option **B**

151. A function can be used in an expression, whereas a subroutine cannot be.

A. True
B. False

Answer: Option **A**

152. Which one of the following classes are present *System.Collections.Generic* namespace?

1. *Stack*
2. *Tree*
3. *SortedDictionary*
4. *SortedArray*

A. 1 and 2 only
B. 2 and 4 only

C. 1 and 3 only
D. All of the above
E. None of the above

Answer: Option **C**

153. Which of the following statements are valid about generics in .NET Framework?

1. Generics is a language feature.
2. We can create a generic class, however, we cannot create a generic interface in C#.NET.
3. Generics delegates are not allowed in C#.NET.
4. Generics are useful in collection classes in .NET framework.
5. None of the above

A. 1 and 2 Only
B. 1, 2 and 3 Only
C. 1 and 4 Only
D. All of the above
E. None of the above

Answer: Option **C**

154. Which of the following statements is valid about generic procedures in C#.NET?

A. All procedures in a Generic class are generic.

B. Only those procedures labeled as Generic are generic.

C. Generic procedures can take at the most one generic parameter.

D. Generic procedures must take at least one type parameter.

E. None of the above.

Answer: Option **D**

155. Which of the following statements is valid about advantages of generics?

A. Generics shift the burden of type safety to the programmer rather than compiler.
B. Generics require use of explicit type casting.
C. Generics provide type safety without the overhead of multiple implementations.
D. Generics eliminate the possibility of run-time errors.
E. None of the above.

Answer: Option **C**

Formatted: Font: Times New Roman, 12 pt