01 Introduction to C# and Data Types  
Understanding Data Types  
Test your Knowledge  
1. What type would you choose for the following “numbers”?  
A person’s telephone number  
A person’s height  
A person’s age  
A person’s gender (Male, Female, Prefer Not To Answer)  
A person’s salary  
A book’s ISBN  
A book’s price  
A book’s shipping weight  
A country’s population  
The number of stars in the universe = 200 billion trillion stars  
The number of employees in each of the small or medium businesses in the  
United Kingdom (up to about 50,000 employees per business)

2. What are the difference between value type and reference type variables?

A Value Type holds the data within its own memory allocation and a Reference Type contains a pointer to another memory location that holds the real data. Reference Type variables are stored in the heap while Value Type variables are stored in the stack.

What is boxing and unboxing?

//ability to convert a value type to a reference type and vice versa

Boxing is the process of converting a [value type](https://docs.microsoft.com/en-us/dotnet/csharp/language-reference/builtin-types/value-types) to the type object or to any interface type implemented by this value type. When the common language runtime (CLR) boxes a value type, it wraps the value inside a [System.Object](https://docs.microsoft.com/en-us/dotnet/api/system.object) instance and stores it on the managed heap. Unboxing extracts the value type from the object. Boxing is implicit; unboxing is explicit. The concept of boxing and unboxing underlies the C# unified view of the type system in which a value of any type can be treated as an object.

3. What is meant by the terms managed resource and unmanaged resource in .NET

Managed resources are those that are pure .NET code and managed by the runtime and are under its direct control.

Unmanaged resources are those that are not. File handles, pinned memory, COM objects, database connections etc.

4. What’s the purpose of Garbage Collector in .NET?

.NET's garbage collector manages the allocation and release of memory for your application. Each time you create a new object, the common language runtime allocates memory for the object from the managed heap. As long as address space is available in the managed heap, the runtime continues to allocate space for new objects. However, memory is not infinite. Eventually the garbage collector must perform a collection in order to free some memory. The garbage collector's optimizing engine determines the best time to perform a collection, based upon the allocations being made. When the garbage collector performs a collection, it checks for objects in the managed heap that are no longer being used by the application and performs the necessary operations to reclaim their memory.

Playing with Console App  
Modify your console application to display a different message. Go ahead and  
intentionally add some mistakes to your program, so you can see what kinds of error messages you get from the compiler. The more familiar you are with these messages, and what causes them, the better you'll be at diagnosing problems in your programs that you / didn't/ intend to add!  
Using just the ReadLine and WriteLine methods and your current knowledge of variables, you can have the user pass in quite a few bits of information. Using this approach, create a console application that asks the user a few questions and then generates some custom output for them. For instance, your program could generate their "hacker name" by asking them their favorite color, their astrology sign, and their street address number. The result might be something like "Your hacker name is RedGemini480."

Practice number sizes and ranges

1. Create a console application project named /02UnderstandingTypes/ that outputs the number of bytes in memory that each of the following number types uses, and the minimum and maximum values they can have: sbyte, byte, short, ushort, int, uint, long, ulong, float, double, and decimal.  
Composite Formatting to learn how to align text in a console application

2. Write program to enter an integer number of centuries and convert it to years, days, hours,minutes, seconds, milliseconds, microseconds, nanoseconds. Use an appropriate data type for every data conversion. Beware of overflows!  
Input: 1  
Output: 1 centuries = 100 years = 36524 days = 876576 hours = 52594560 minutes  
= 3155673600 seconds = 3155673600000 milliseconds = 3155673600000000  
microseconds = 3155673600000000000 nanoseconds  
Input: 5  
Output: 5 centuries = 500 years = 182621 days = 4382904 hours = 262974240 minutes = 15778454400 seconds = 15778454400000 milliseconds = 15778454400000000  
microseconds = 15778454400000000000 nanoseconds

Controlling Flow and Converting Types  
Test your Knowledge  
1. What happens when you divide an int variable by 0?

**System.DivideByZeroException:** 'Attempted to divide by zero.'

2. What happens when you divide a double variable by 0?

Output: ∞

3. What happens when you overflow an int variable, that is, set it to a value beyond it range?

Error: Cannot implicitly convert type ‘long’ to ‘int’. An explicit conversion exitsts(are you missing a cast?)

Ex: change int x=1000000000000 to int x = (int)1000000000000

4. What is the difference between x = y++; and x = ++y;?

 ++x happens prior to assignment (pre-increment), but x++happens after assignment (post-increment). x++ executes the statement and then increments the value. ++x increments the value and then executes the statement.

5. What is the difference between break, continue, and return when used inside a loop statement?

**break statement**

The break statement is used to terminate the loop or statement in which it present. After that, the control will pass to the statements that present after the break statement, if available. If the break statement present in the nested loop, then it terminates only those loops which contains break statement.

**continue statement**

This statement is used to skip over the execution part of the loop on a certain condition. After that, it transfers the control to the beginning of the loop. Basically, it skips its following statements and continues with the next iteration of the loop.

**return statement**

This statement terminates the execution of the method and returns the control to the calling method. It returns an optional value. If the type of method is void, then the return statement can be excluded.

6. What are the three parts of a for statement and which of them are required?

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

{

Console.WriteLine(i);

}

**Statement 1 sets a variable before the loop starts (int i = 0).**

**Statement 2 defines the condition for the loop to run (i must be less than 5). If the condition is true, the loop will start over again, if it is false, the loop will end.**

**Statement 3 increases a value (i++) each time the code block in the loop has been executed.**

7. What is the difference between the = and == operators?

= is assignment operator

== is equal to

8. Does the following statement compile? for ( ; true; ) ;

yes

9. What does the underscore \_ represent in a switch expression?

* The underscore (**\_**) character replaces the **default**keyword to signify that it should match anything if reached.
* The bodies are now expressions instead of statements. The selected body becomes the switch expression’s result.

10. What interface must an object implement to be enumerated over by using the foreach statement?

The IEnumerable interface provides support for the foreach iteration. IEnumerable requires that you implement the GetEnumerator method.

Practice loops and operators  
1. FizzBuzz is a group word game for children to teach them about division. Players take turns to count incrementally, replacing any number divisible by three with the word /fizz/, any number divisible by five with the word /buzz/, and any number divisible by both with /fizzbuzz/.Create a console application in Chapter03 named Exercise03 that outputs a simulated FizzBuzz game counting up to 100. The output should look something like the following screenshot:  
What will happen if this code executes?  
int max = 500;  
for (byte i = 0; i < max; i++)  
{  
WriteLine(i);  
}

Create a console application and enter the preceding code. Run the console application and view the output. What happens?

It’s infinite loop.

What code could you add (don’t change any of the preceding code) to warn us about the problem?

The range of byte is 0-255, so if assign byte to 255 or more, it will cause overflow. So add a break statement when I is overflown.

Your program can create a random number between 1 and 3 with the following code:  
int correctNumber = new Random().Next(3) + 1;

Write a program that generates a random number between 1 and 3 and asks the user to guess what the number is. Tell the user if they guess low, high, or get the correct answer. Also, tell the user if their answer is outside of the range of numbers that are valid guesses (less than 1 or more than 3). You can convert the user's typed answer from a string to an  
int using this code:

of 4



02 Arrays and Strings  
Test your Knowledge  
1. When to use String vs. StringBuilder in C# ?

* If a string is going to remain constant throughout the program, then use String class object because a String object is immutable.
* If a string can change (example: lots of logic and operations in the construction of the string) then using a StringBuilder is the best option.

2. What is the base class for all arrays in C#?

The Array class is the base class for all the arrays in C#. It is defined in the System namespace. The Array class provides various properties and methods to work with arrays.

3. How do you sort an array in C#?

sWe can sort a one-dimensional array in two ways, using Array.Sort() method and using LINQ query.

4. What property of an array object can be used to get the total number of elements in an array?

**Array.Length Property** is used to get the total number of elements in all the dimensions of the Array. Basically, the length of an array is the total number of the elements which is contained by all the dimensions of that array.

5. Can you store multiple data types in System.Array?

No, we cannot store multiple datatype in an Array, we can store similar datatype only in an Array.

6. What’s the difference between the System.Array.CopyTo() and System.Array.Clone()?

The **Clone()** method returns a new array object containing all the elements in the original array. This method creates a copy of an array as an object, therefore needs to be **cast to** the actual array type before it can be used to do very much. The clone is of the same Type as the original Array.

The **CopyTo()** method copies the elements into another existing array. It copies the elements of one array to another **pre-existing** array starting from a given index (usually 0).

Both perform a **shallow copy** . Shallow copying is creating a new object and then copying the non static fields of the current object to the new object. If the field is a value type, a **bit by bit** copy of the field is performed. If the field is a reference type, the reference is copied but the referred object is not, therefore the original object and its clone refer to the same object. A shallow copy of an object is a **new object** whose instance variables are identical to the old object.

Practice Arrays  
1. Copying an Array  
Write code to create a copy of an array. First, start by creating an initial array. (You can use whatever type of data you want.) Let’s start with 10 items. Declare an array variable and assign it a new array with 10 items in it. Use the things we’ve discussed to put some values in the array.  
Now create a second array variable. Give it a new array with the same length as the first. Instead of using a number for this length, use the Lengthproperty to get the size of the original array.  
Use a loop to read values from the original array and place them in the new array. Also print out the contents of both arrays, to be sure everything copied correctly.

2. Write a simple program that lets the user manage a list of elements. It can be a grocery list, "to do" list, etc. Refer to Looping Based on a Logical Expression if necessary to see how to implement an infinite loop. Each time through the loop, ask the user to perform an operation, and then show the current contents of their list. The operations available should be Add, Remove, and Clear. The syntax should be as follows:  
+ some item  
- some item  
--

Your program should read in the user's input and determine if it begins with a “+” or “-“ or if it is simply “—“ . In the first two cases, your program should add or remove the string given ("some item" in the example). If the user enters just “—“ then the program should clear the current list. Your program can start each iteration through its loop with the following instruction:

Console.WriteLine("Enter command (+ item, - item, or -- to clear)):");

3. Write a method that calculates all prime numbers in given range and returns them as array of integers  
static int[] FindPrimesInRange(startNum, endNum)  
{  
}

4. Write a program to read an array of n integers (space separated on a single line) and an integer k, rotate the array right k times and sum the obtained arrays after each rotation as shown below. After r rotations the element at position I goes to position (I + r) % n. The sum[] array can be calculated by two nested loops: for r = 1 ... k; for I = 0 ... n-1.  
Input Output Comments  
3 2 4 -1 3 2 5 6 rotated1[] = -1 3 2 4  
2 rotated2[] = 4 -1 3 2  
sum[] = 3 2 5 6  
1 2 3 4 5 12 10 8 6 9 rotated1[] = 5 1 2 3 4  
3 rotated2[] = 4 5 1 2 3  
rotated3[] = 3 4 5 1 2  
sum[] = 12 10 8 6 9

5. Write a program that finds the longest sequence of equal elements in an array of integers. If several longest sequences exist, print the leftmost one.  
Input Output  
2 1 1 2 3 3 2 2 2 1 2 2 2  
1 1 1 2 3 1 3 3 1 1 1  
4 4 4 4 4 4 4 4  
0 1 1 5 2 2 6 3 3 1 1

7. Write a program that finds the most frequent number in a given sequence of numbers. In case of multiple numbers with the same maximal frequency, print the leftmost of them

Input Output  
4 1 1 4 2 3 4 4 1 2 4 9 3 The number 4 is the most frequent (occurs 5 times)  
7 7 7 0 2 2 2 0 10 10 10 The numbers 2, 7 and 10 have the same maximal  
frequence (each occurs 3 times). The leftmost of them is 7.

Practice Strings

1. Write a program that reads a string from the console, reverses its letters and prints the result back at the console.  
Write in two ways Convert the string to char array, reverse it, then convert it to string again Print the letters of the string in back direction (from the last to the first) in a for-loop  
Input Output  
sample elpmas  
24tvcoi92 29iocvt42

2. Write a program that reverses the words in a given sentence without changing the punctuation and spaces. Use the following separators between the words: . , : ; = ( ) & [ ] " ' \ / ! ? (space).  
All other characters are considered part of words, e.g. C++, a+b, and a77 are  
considered valid words.  
The sentences always start by word and end by separator.  
C# is not C++, and PHP is not Delphi!  
Delphi not is PHP, and C++ not is C#!  
The quick brown fox jumps over the lazy dog /Yes! Really!!!/.  
Really Yes dog lazy the over jumps fox brown /quick! The!!!/.

3. Write a program that extracts from a given text all palindromes, e.g. “ABBA”, “lamal”, “exe” and prints them on the console on a single line, separated by comma and space.Print all unique palindromes (no duplicates), sorted  
Hi,exe? ABBA! Hog fully a string: ExE. Bob  
a, ABBA, exe, ExE

4. Write a program that parses an URL given in the following format:  
[protocol]://[server]/[resource]  
The parsing extracts its parts: protocol, server and resource.  
The [server] part is mandatory.  
The [protocol] and [resource] parts are optional.  
https://www.apple.com/iphone  
[protocol] = "https"  
[server] = "www.apple.com"  
[resource] = "iphone"  
ftp://www.example.com/employee  
[protocol] = "ftp"  
[server] = "www.example.com"  
[resource] = "employee"  
https://google.com  
[protocol] = "https"  
[server] = "google.com"  
[resource] = ""  
www.apple.com  
[protocol] = ""  
[server] = "www.apple.com"  
[resource] = ""