

# C# Fundamentals

Preparation 4 – functions and  
methods

# Last time

- We learned how to declare a variable and an array.

# Today

- We will focus on functions, and how they are constructed.

# Your application

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

```
namespace MyFirstApplication
```

```
{
```

```
class Program
```

```
{
```

```
static void Main(string[] args)
```

```
{
```

```
}
```

```
}
```

```
}
```

We will mainly  
work with this  
part of the code!

# Some math

- A function can take **several** parameters, **but**, it can only return **one** value.
- A method can take **several** parameters, but, it **cannot** return a value.

# Creating a function [Demo]

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

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

        }

        static int AddNumbers(int number1, int number2)
        {
            return number1 + number2;
        }
    }
}
```

# Methods

- A method is behaving as a function, but, it does not return a value.
- Declared using **void** keyword.

# Creating a method [Demo]

```
class Program
{
    static void Main(string[] args)
    {
        DisplayHello("Allan");
    }
    static void DisplayHello(string name)
    {
        Console.WriteLine("Hello" + name);
    }
}
```



# Functions and Methods

- Functions and Methods are as **paragraphs** in the English language.
- When you write a paragraph, it's important to focus on only one **main** idea.

# Functions and Methods

- Functions and Methods should be **specific**. A function should either *decide* or *do*!
- Compare a program to an essay, or a team of people.



# Functions and Methods

- An essay consist of an **introduction** and **supporting paragraphs**.
- The introduction is the method that decides.
- The supporting paragraphs are methods that are specialized to do one main thing.

# Functions and Methods

- In a team, you have a chief that should instruct other team member.
- When creating a method, it should only have one role: **a decider** or **a doer**.

# Summary

- Today we have looked at **functions** and **methods**.

# Thank you for watching!

- Next lecture is at {date}.
- Any questions, please let me know:
  - [vcs@clizware.net](mailto:vcs@clizware.net)
  - [vcs@artemlos.net](mailto:vcs@artemlos.net)