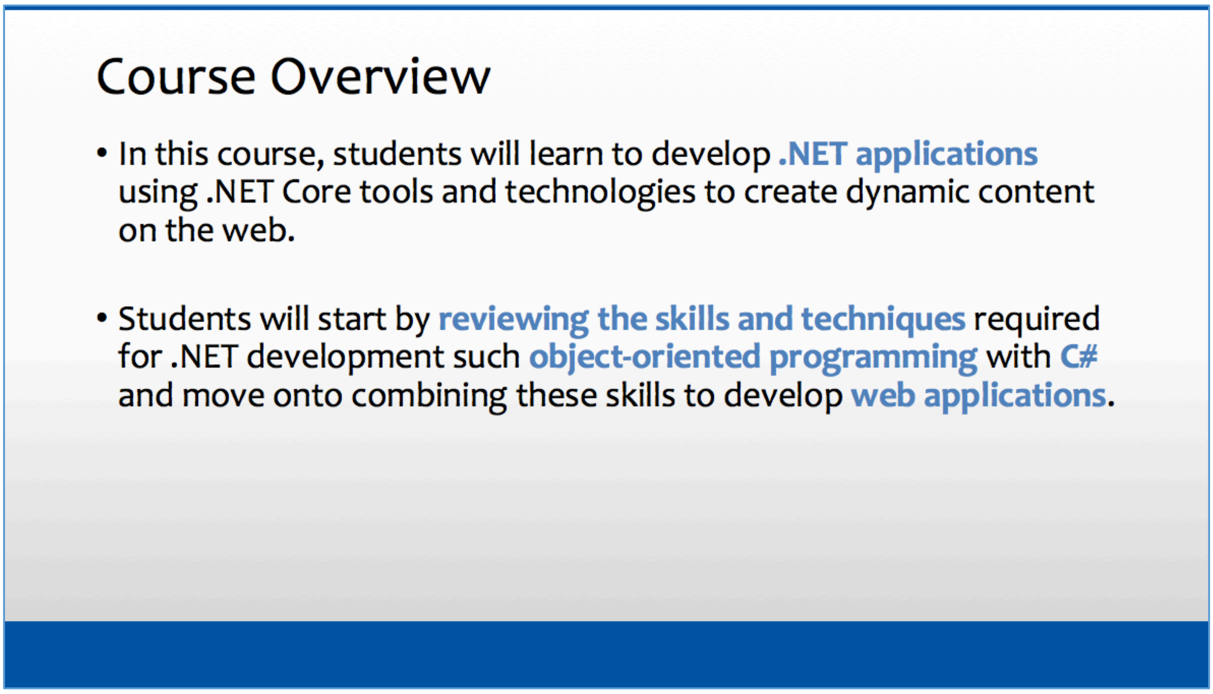
Web Development .NET
Week One: Getting Started

Bahamas Technical & Vocational Institute

EST 1980

Learn, Grow, Achieve

Web Development .NET



Course Overview

- In this course, students will learn to develop **.NET applications** using .NET Core tools and technologies to create dynamic content on the web.
- Students will start by **reviewing the skills and techniques** required for .NET development such **object-oriented programming** with **C#** and move onto combining these skills to develop **web applications**.

Expectations

- **Live Zoom Sessions**
  Fridays 12pm (all sessions will be recorded)

- **Grading**

Attendance / Participation	5%
Lab Exercises (Activities, etc.)	25%
Assignments	25%
Unit Tests	15%
Mid Term Evaluation	10%
Final Evaluation	20%

- **Academic Policies**

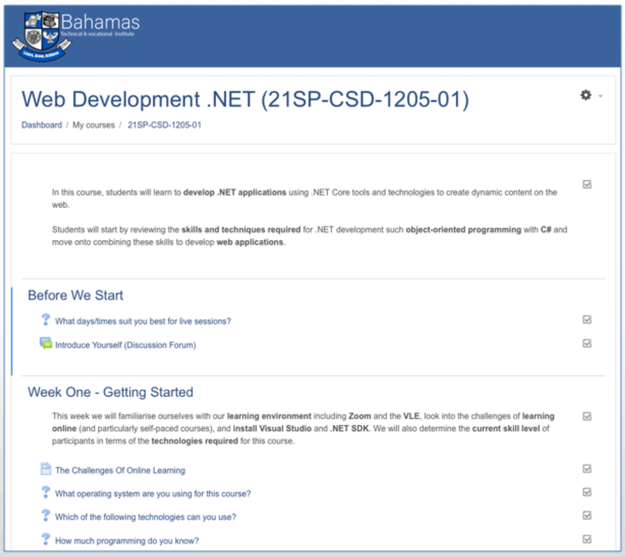
Always
do your best.
What you plant
now, you will
harvest later.
Og Mandino

FUN
– IS –
GOOD.
DR. SEUSS

MISTAKES
are proof
that you are
TRYING

Using The VLE

- All course materials, details of required tasks, assignments, etc. can be found on the VLE.
- All announcements will be posted on the VLE in addition to sent by e-mail.
- You should log into the VLE between 4-7 days a week.



Web Development .NET (21SP-CSD-1205-01)

Dashboard / My courses / 21SP-CSD-1205-01

In this course, students will learn to **develop .NET applications** using .NET Core tools and technologies to create dynamic content on the web.

Students will start by reviewing the **skills and techniques required** for .NET development such **object-oriented programming** with **C#** and move onto combining these skills to develop **web applications**.

Before We Start

- What days/times suit you best for live sessions? ☑
- Introduce Yourself (Discussion Forum) ☑

Week One - Getting Started

This week we will familiarise ourselves with our **learning environment** including **Zoom** and the **VLE**, look into the challenges of **learning online** (and particularly self-paced courses), and **install Visual Studio** and **.NET SDK**. We will also determine the **current skill level** of participants in terms of the **technologies required** for this course.

- The Challenges Of Online Learning ☑
- What operating system are you using for this course? ☑
- Which of the following technologies can you use? ☑
- How much programming do you know? ☑

Getting Started

- Students will start by reviewing the **skills and techniques** required for .NET development such **object-oriented programming** with **C#** and move onto combining these skills to develop **web applications**.

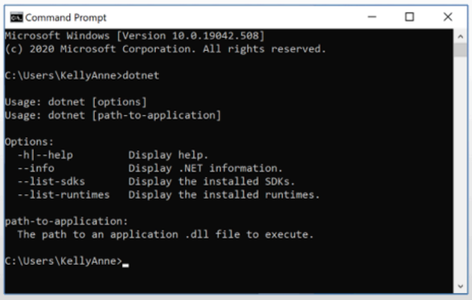
- Student Surveys - Student Hardware and Software
  Experience with HTML, CSS, Bootstrap
  Programming In C#, Visual Basic or Other
  Familiarity With Visual Studio

Lab Exercise:

- Download and install the **.NET SDK** (version 5 recommended).
- Download and install **Visual Studio 2019** (Community is fine).

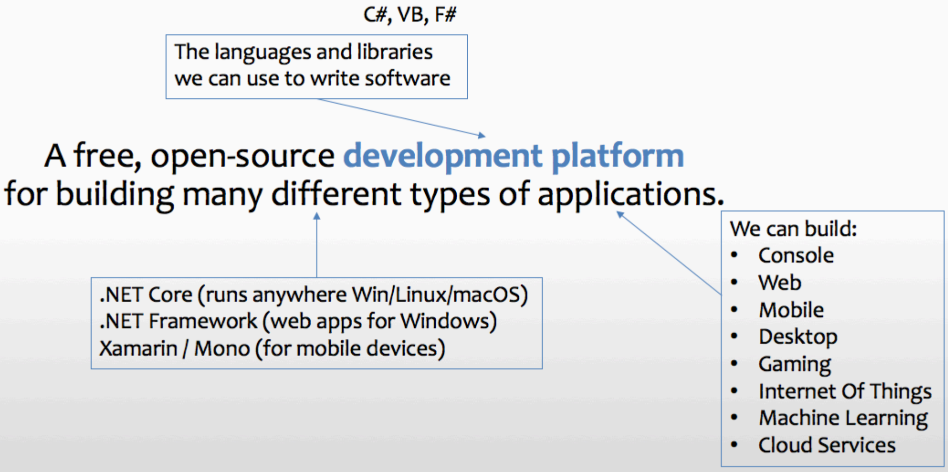
- Check that .NET has installed correctly by typing **dotnet** into the Command Prompt.

- Submit a screenshot of your Command Prompt window to demonstrate that .NET has been installed correctly.



```
Microsoft Windows [Version 10.0.19042.508]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\KellyAnne>dotnet

Usage: dotnet [options]
Usage: dotnet [path-to-application]

Options:
  -h|--help         Display help.
  --info            Display .NET information.
  --list-sdks       Display the installed SDKs.
  --list-runtimes   Display the installed runtimes.

path-to-application:
  The path to an application .dll file to execute.

C:\Users\KellyAnne>_
```

What Is .NET?



C#, VB, F#

The languages and libraries we can use to write software

A free, open-source **development platform** for building many different types of applications.

.NET Core (runs anywhere Win/Linux/macOS)
.NET Framework (web apps for Windows)
Xamarin / Mono (for mobile devices)

We can build:

- Console
- Web
- Mobile
- Desktop
- Gaming
- Internet Of Things
- Machine Learning
- Cloud Services

History Of .NET

- The original .NET Framework was first released in 2002.
  Many updates and functionality have been added since.

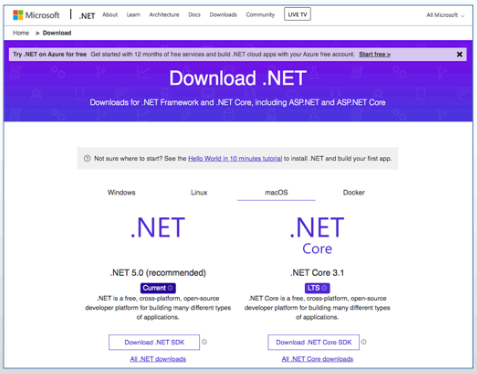
- .NET Core was introduced in 2014 for cross-platform compatibility.
  Allows us to write software that runs on macOS, Linux, and Windows.

- The original .NET Framework has been maintained.
  But new features and improvements are reserved for .NET Core.
  Note that “Core” was later removed from the name.

- The latest version is currently .NET 5.

How To Install .NET

- The easiest way is to install Visual Studio 2019.
- You can also download and install .NET manually with the command line.
- .NET can actually be used on any platform or operating system with any editor.



Try .NET on Azure for free Get started with 12 months of free services and build .NET cloud apps with your Azure free account.

Download .NET

Downloads for .NET Framework and .NET Core, including ASP.NET and ASP.NET Core

Not sure where to start? See the Hello World in 10 minutes tutorial to install .NET and build your first app.

.NET 5.0 (recommended)

.NET is a free, cross-platform, open-source developer platform for building many different types of applications.

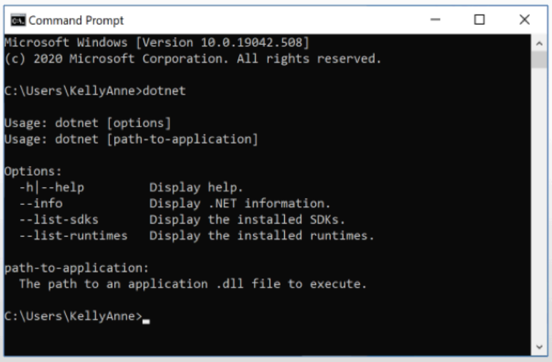
.NET Core 3.1

.NET Core is a free, cross-platform, open-source developer platform for building many different types of applications.

https://dotnet.microsoft.com/download

How To Install .NET (cont.)

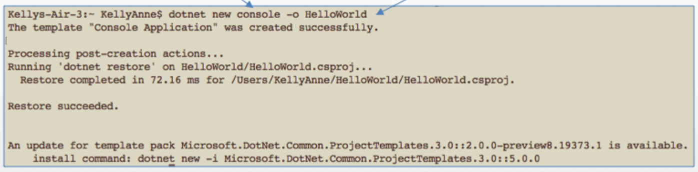
- You can check that .NET is installed correctly on your computer by running **dotnet** from the command line.



```
Microsoft Windows [Version 10.0.19042.508]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\KellyAnne>dotnet

Usage: dotnet [options]
Usage: dotnet [path-to-application]

Options:
  -h|--help         Display help.
  --info            Display .NET information.
  --list-sdks       Display the installed SDKs.
  --list-runtimes   Display the installed runtimes.

path-to-application:
  The path to an application .dll file to execute.

C:\Users\KellyAnne>_
```

Creating Our First .NET Application

- The following code demo will introduce the concept of creating a typical “Hello World” .NET console application.

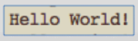


Template

Create new folder (directory)

```
Kellys-Air-3:~ KellyAnne$ dotnet new console -o HelloWorld
The template "Console Application" was created successfully.

Processing post-creation actions...
Running 'dotnet restore' on HelloWorld/HelloWorld.csproj...
  Restore completed in 72.16 ms for /Users/KellyAnne/HelloWorld/HelloWorld.csproj.

Restore succeeded.


An update for template pack Microsoft.DotNet.Common.ProjectTemplates.3.0::2.0.0-preview8.19373.1 is available.
    install command: dotnet new -i Microsoft.DotNet.Common.ProjectTemplates.3.0::5.0.0
```

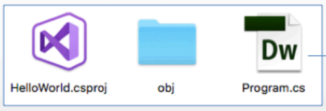
- From inside the directory, we can type **dotnet run** to run our app. We can do this from macOS or Windows.

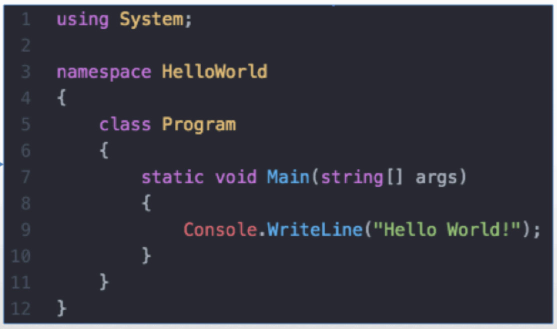


```
Hello World!
```

Creating Our First .NET Application (cont.)

- We can see the following files inside the created folder.

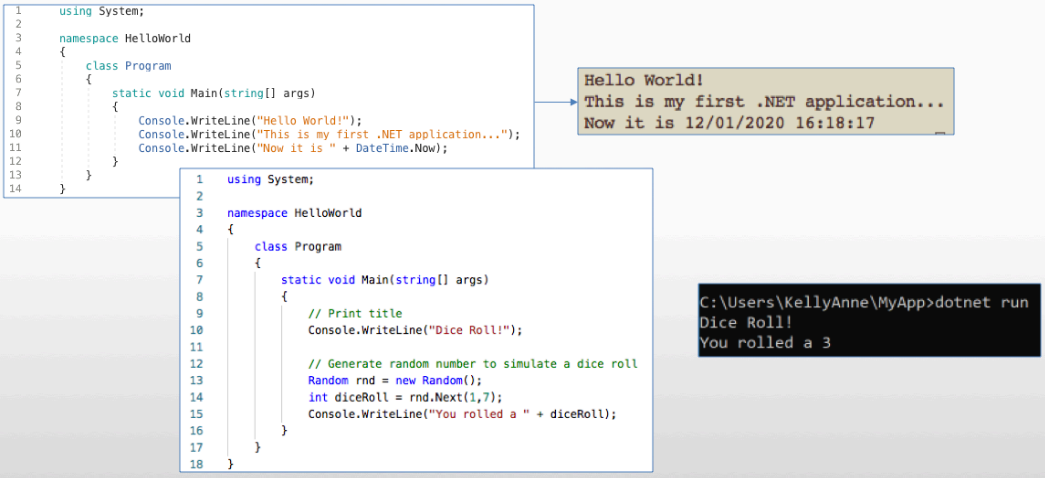




```
using System;

namespace HelloWorld
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hello World!");
        }
    }
}
```

Program.cs

Making Changes To Program.cs



```
using System;

namespace HelloWorld
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hello World!");
            Console.WriteLine("This is my first .NET application...");
            Console.WriteLine("Now it is " + DateTime.Now);
        }
    }
}
```

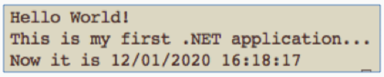
```
Hello World!
This is my first .NET application...
Now it is 12/01/2020 16:18:17
```

```
using System;

namespace HelloWorld
{
    class Program
    {
        static void Main(string[] args)
        {
            // Print title
            Console.WriteLine("Dice Roll!");

            // Generate random number to simulate a dice roll
            Random rnd = new Random();
            int diceRoll = rnd.Next(1,7);
            Console.WriteLine("You rolled a " + diceRoll);
        }
    }
}
```

```
C:\Users\KellyAnne\MyApp>dotnet run
Dice Roll!
You rolled a 3
```

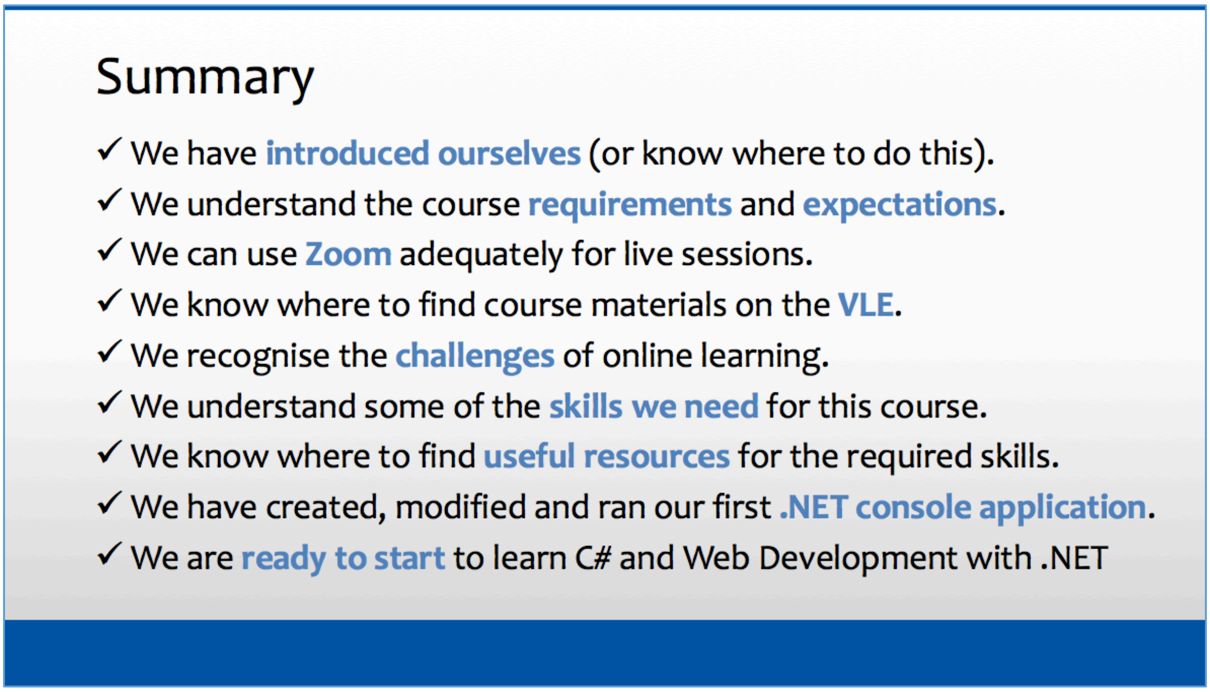
Lab Exercise: Create A .NET Console App

- Follow the instructions to create a .NET console application. It should print out your full name and the current date/time.



```
Hello World!
This is my first .NET application...
Now it is 12/01/2020 16:18:17
```

- Deliverables:
  - A screenshot of your Program.cs code showing the code you wrote.
  - A screenshot of Command Prompt once your application has run.



Summary

✓ We have **introduced ourselves** (or know where to do this).

✓ We understand the course **requirements** and **expectations**.

✓ We can use **Zoom** adequately for live sessions.

✓ We know where to find course materials on the **VLE**.

✓ We recognise the **challenges** of online learning.

✓ We understand some of the **skills we need** for this course.

✓ We know where to find **useful resources** for the required skills.

✓ We have created, modified and ran our first **.NET console application**.

✓ We are **ready to start** to learn C# and Web Development with .NET