INTRODUCTION TO PROGRAMMING

USING C#

Lecture 2: Hello World!, Our first program

CONTENTS:

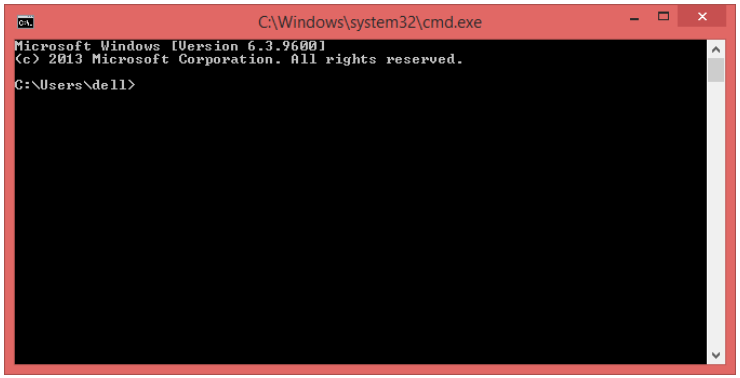
- Hello World!
- Console Applications.
- Printing text to the console.
- Halting execution.
- Printing line of text to the console.
- Variables and constants.
- Data types.
- Reading text from the user.
- Users and Programmers.
- Console colors.
- Getting and setting properties.
- Demo (chat with pc)

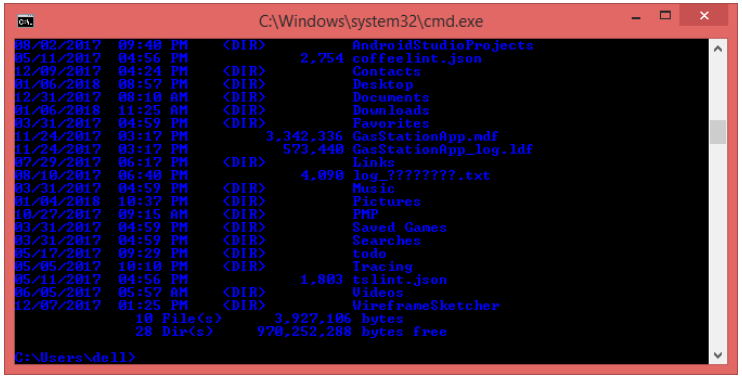
HELLO WORLD!

It is a tradition in programming books and courses to start with a simple example that prints the text “Hello World!”

CONSOLE APPLICATIONS

- An application with the simplest UI ever, Just text.
- You can pass arguments to console app by appending them to command line separating them with a space.
- It’s an option for programs that do single task with no much Interactivity from the user.





PRINTING TEXT TO THE CONSOLE

```
Console.Write("Hello World!");
```

HALTING EXECUTION

```
Console.Write("Hello World!");
Console.ReadLine();
```

PRINTING LINE OF TEXT TO THE CONSOLE

```
Console.Write("Hello World!");
Console.Write("Hello World!");
Console.ReadLine();
```

PRINTING LINE OF TEXT TO THE CONSOLE

```
Console.WriteLine("Hello World!");
Console.WriteLine("Hello World!");
Console.ReadLine();
```

VARIABLES AND CONSTANTS

- A variable is a named place in the memory that hold some value
- A variable value can change, Constant value can't be changed.
- You can define new variable using the form

Datatype *VariableName* [=value];

VARIABLES AND CONSTANTS

- **To define new constant use the following form**

const Datatype *constantName* [=value];

VARIABLES AND CONSTANTS

Variable and constant naming roles :

- **Must start with a letter.**
- **Can’t contain white space or special characters.**
- **Not a reserved word.**

DATA TYPES

Data can exists in many types

- **Numerical (integer, decimal, float …)**
- **Strings**
- **Date and time**
- **Structures**

DATA TYPES

Numerical data types / Integral types :

Type	Range	Size
sbyte	-128 to 127	Signed 8-bit integer
byte	0 to 255	Unsigned 8-bit integer
char	U+0000 to U+ffff	Unicode 16-bit character
short	-32,768 to 32,767	Signed 16-bit integer
ushort	0 to 65,535	Unsigned 16-bit integer
int	-2,147,483,648 to 2,147,483,647	Signed 32-bit integer
uint	0 to 4,294,967,295	Unsigned 32-bit integer
long	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807	Signed 64-bit integer
ulong	0 to 18,446,744,073,709,551,615	Unsigned 64-bit integer

https://docs.microsoft.com/en-us/dotnet/csharp/language-reference/keywords/value-types

DATA TYPES

Numerical data types / Floating-Point types :

Type	Approximate range	Precision
float	±1.5e−45 to ±3.4e38	7 digits
double	±5.0e−324 to ±1.7e308	15-16 digits

https://docs.microsoft.com/en-us/dotnet/csharp/language-reference/keywords/value-types

DATA TYPES

Numerical data types / decimal:

Type	Approximate Range	Precision	.NET Framework type
`decimal`	(-7.9 x 10^{28} to 7.9 x 10^{28}) / (10^{0} to 10^{28})	28-29 significant digits	System.Decimal

https://docs.microsoft.com/en-us/dotnet/csharp/language-reference/keywords/value-types

DATA TYPES

Boolean data types:

```
bool IsExists = false;
bool CanDeleteItems = true;
```

DATA TYPES

Enumeration data types:

```
enum Direction
{
    Left,
    Right
}
```

```
enum Gender
{
    Female,
    Male
}
```

```
enum SwitchStat
{
    On,
    Off
}
```

DATA TYPES

Strings data types:

```
string MyString = "Hello World!";

MyString = "Hello\nWorld!";

MyString = "Hello \"World\"!";
```

READING TEXT FROM THE USER

```
string s = "";
s = Console.ReadLine();
```

READING TEXT FROM THE USER

```
Console.WriteLine("What is your name?");
string s = "";
s = Console.ReadLine();
Console.WriteLine("Hi " + s + ", How are you?");
Console.ReadLine();
```

USERS AND PROGRAMMERS

- **Programmer is you, who writes programs.**
- **User is the one who use your program**

CONSOLE COLORS

You can change the text color or the background color using something like the following code:

```
Console.ForegroundColor = ConsoleColor.Blue;
Console.BackgroundColor = ConsoleColor.Yellow;
```

GETTING AND SETTING PROPERTIES

```
VariableName = object.ProperityName;

object.ProperityName = value | variableName
```

DEMO

Chatting with PC

NEXT

- **Mathematics operations.**
- **Conditions and loops.**