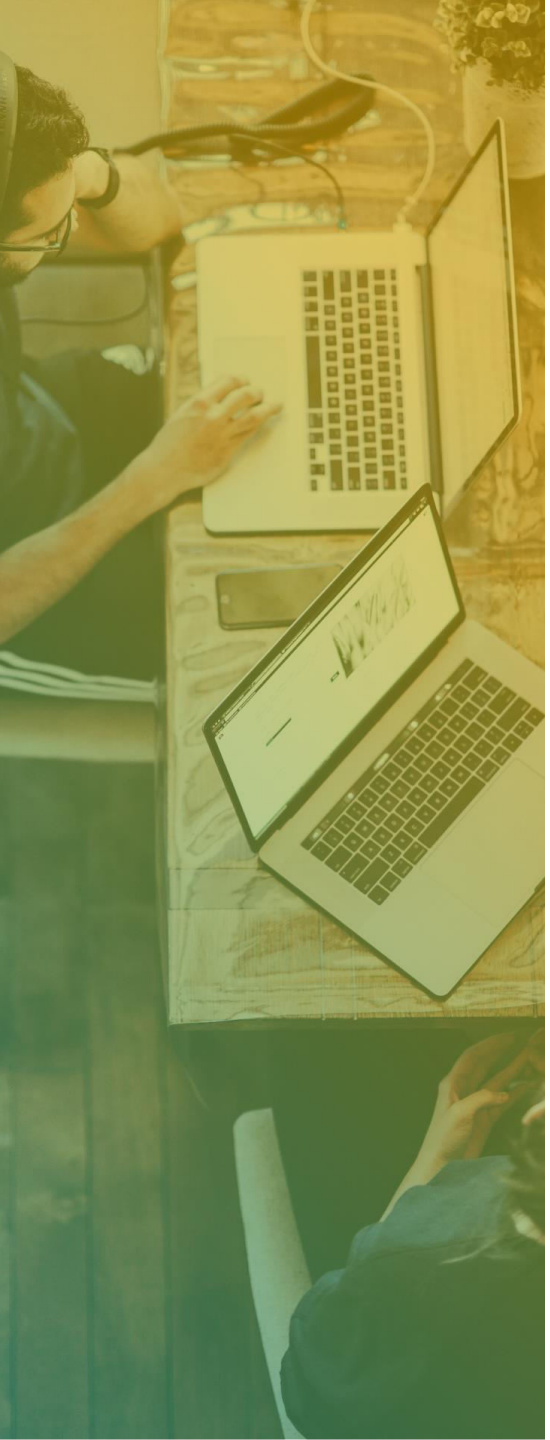


C# Overview

Introduction

C# & .NET Development Fundamentals

Illés Horpácsi

CUBIX
INSTITUTE OF TECHNOLOGY

Course Overview

Week	Topic	
1	C# Overview	
2	Let's practice!	LIVE!
3	Object-oriented Programming 1.	
4	Object-oriented Programming 2.	LIVE!
5	.NET Ecosystem 1.	
6	.NET Ecosystem 2.	LIVE!
7	Let's practice!	LIVE!
8	.NET in practice 1.	
9	.NET in practice 2.	LIVE!
10	Let's practice!	LIVE!
11	Final exam	LIVE!

Live!

- Your questions
- Code review / uploaded solutions
- More examples for topics

- Sessions are recorded
  - BUT more useful if you are there!

Uploaded Solutions

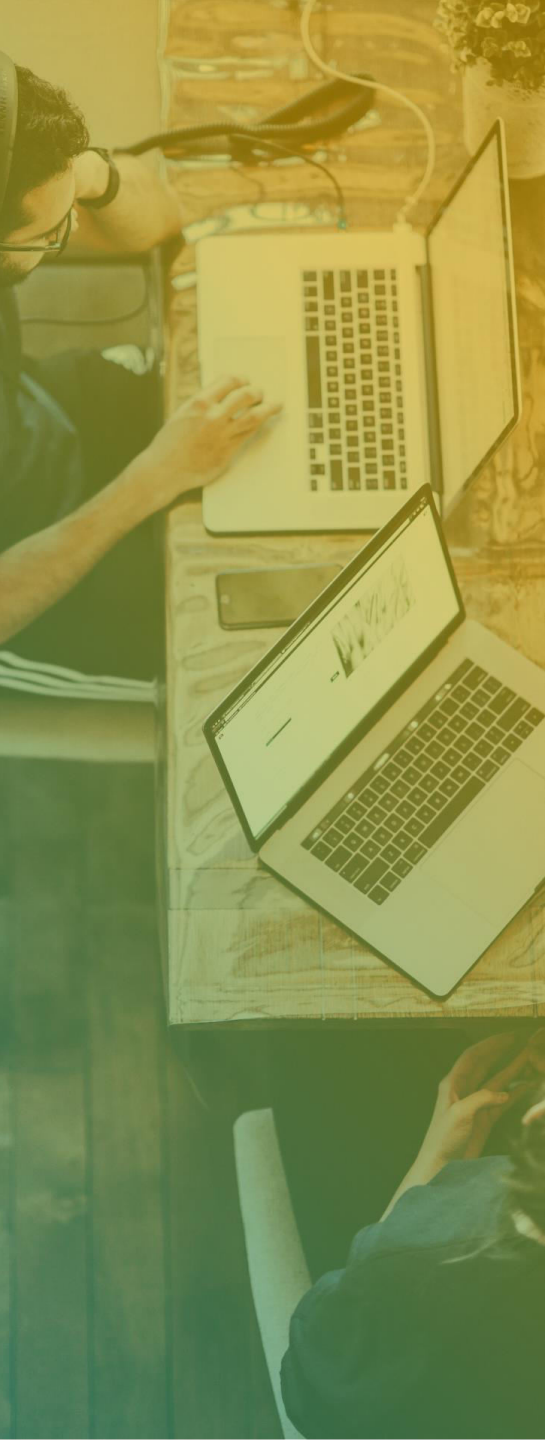
- Not mandatory for certificate
- Use ZIP
- Include: sln / csproj / cs files
- Do NOT include: .vs / bin /obj folders

Final exam

- Not mandatory
- Code + video
- Task: complex software
  - You may choose parts
  - Freedom in technologies

Useful Links

- https://visualstudio.microsoft.com/
- https://www.hackerrank.com/
- https://learn.microsoft.com/en-us/
- https://courses.cubixedu.com/course-list/category/c-sharp-and-dotnet
- https://rosettacode.org/wiki/Hello_world/Text



C# Overview

Types, Variables, User input

C# & .NET Development Fundamentals

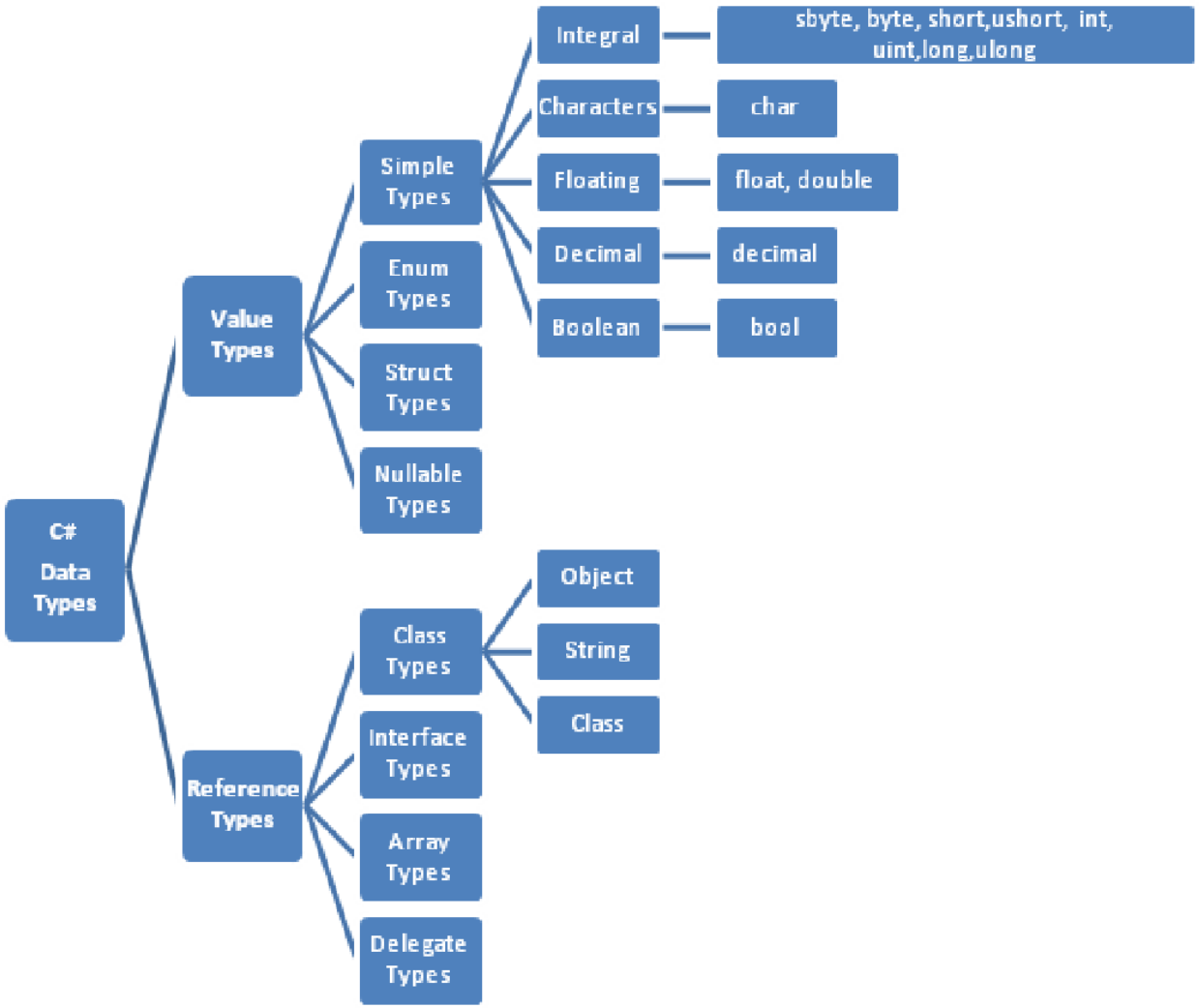
Illés Horpácsi

CUBIX
INSTITUTE OF TECHNOLOGY

Keywords

abstract
as
base
bool
break
byte
case
catch
char
checked
class
const
continue
decimal
default
delegate
do
double
else
enum
event
explicit
extern
false
finally
fixed
float
for
foreach
goto
if
implicit
in
int
interface
internal
is
lock
long
namespace
new
null
object
operator
out
override
params
private
protected
public
readonly
ref
return
sbyte
sealed
short
sizeof
stackalloc
static
string
struct
switch
this
throw
true
try
typeof
uint
ulong
unchecked
unsafe
ushort
using
virtual
void
volatile
while

Variables



Declarations

- Strictly typed language
- Let there be a vaiable containing an intger, and its name shall be „flowers”.
  - int flowers
- Generally:
  - [type] [name]
- Examples
  - string text
  - float myNumber
  - Dog dog
- Usage
  - text = ”Apple”;
  - myNumber = 42;
- Case sensitve! (text <> Text)

CUBIX
INSTITUTE OF TECHNOLOGY

Own Type

> Type Declaration

```
internal class Dog
{
    1 reference
    public void PlayGame()
    {
        Console.WriteLine("I'm running...");
    }
}
```

> Variable Declaration / Usage

```
static void Main(string[] args)
{
    Dog Killer;
    Killer = new Dog();
    Killer.PlayGame();
}
```

Commands

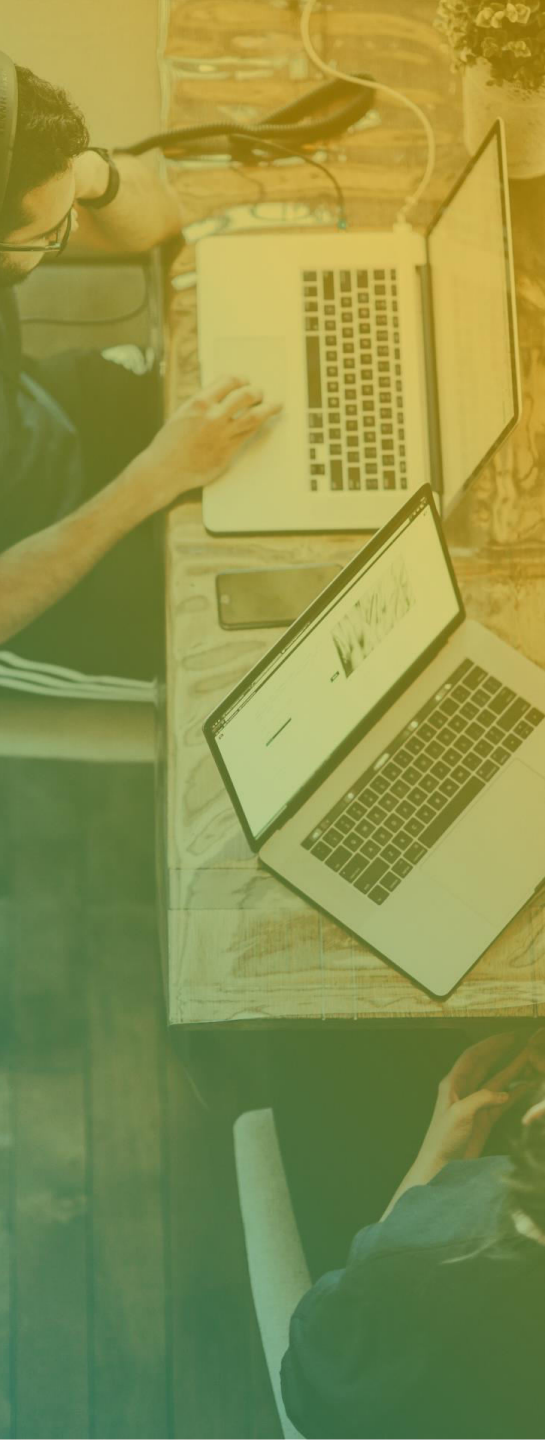
- Ask my dog to play:
  - Killer.PlayGame();
- Command usually has it's parameter(s)
  - int hundred = int.Parse("100");

User Inputs

- Get character code
  - int charcode = Console.Read();
- Any key (not just chars)
  - ConsoleKeyInfo key = Console.Readkey();
- Full line of input
  - string line = Console.ReadLine();

Useful Links

- https://learn.microsoft.com/en-us/dotnet/csharp/language-reference/keywords/
- https://codepattern.net/Blog/post/Data-Types-in-C-sharp



C# Overview

Conditional Sequences, Loops, Exception Handling, Operators

C# & .NET Development Fundamentals

Illés Horpácsi

EUBIX
INSTITUTE OF TECHNOLOGY

If / else

```
If ([condition])
        [command, if condition was true]

If ([condition])
{
        [commands, if condition was true]
}

If ([condition])
{
        [commands, if condition was true]
}
else
{
        [commands, if condition was not true]
}
```

Else if

```
If (x==1)
{
…
}
else if(x==2)
{
…
}
else
{
…
}
```

Ternary conditional operator

[result] = [condition] ? [result if true] : [result if false];

Example:

```
bool valid = true;
...
int num = valid ? 1 : 100;
```

Switch statements

```
switch (expression)
{
        case x:
                // command
        break;
        case y:
                // command
        break;
...
        default:
                // command
        break;

}
```

Loops

- `for (int i=0; i<10; i++) {…}`
- `do {…} while(i<10)`
- `while(i<10) {…}`

- `foreach(int i in ints) {…}`

Exception

```
try
{
        [do your job]
}
catch   //optionally we may get just given type of exceptions
{
        [do something if we failed]
}
finally //optional
{
        [must done failed or not]
}
```

Comments

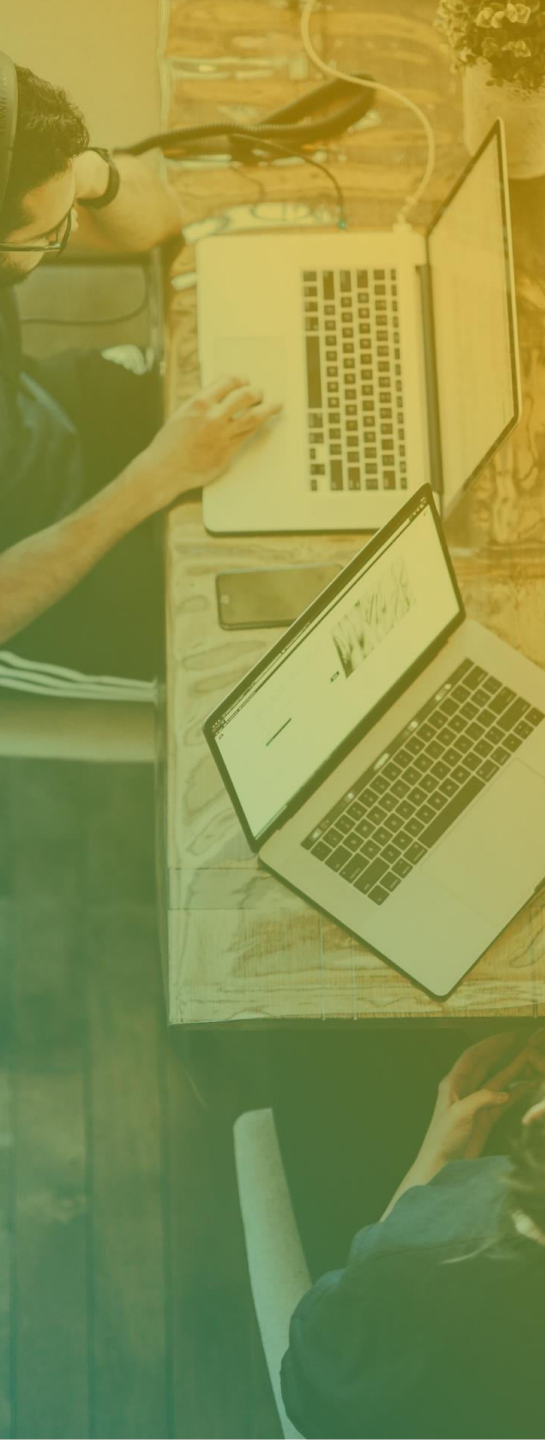
- // one line comment
- /* multiple
  lines
  of
  comment */
- Method comment

```
/// <summary>
/// Add two numebrs
/// </summary>
/// <param name="a">first number</param>
/// <param name="b">second number</param>
/// <returns>sum of the two numbers</returns>
0 references
static int Add(int a, int b)
{
    return a + b;
}
```

CUBIX
INSTITUTE OF TECHNOLOGY

Operators

- Logical operations
  - ||
  - &&
- Mathematical operations
  - +
  - -
  - *
  - /
  - %
- Operators
  - ++
  - --
  - +=
  - -=
  - *=
  - /=
  - %=



C# Overview

HW: MasterMind

C# & .NET Development Fundamentals

Illés Horpácsi

CUBIX
INSTITUTE OF TECHNOLOGY

Tasks

- Our program „thinks” of a number between 0 and 999
- The user must find the secret number
  - Unlimited guesses, validated, numbers only AND only in given range!
- We answer every digit, e.g.:
  - The first number is higher
  - The second number is correct
  - The third number is lower
- The number of attempts is shown at and for the game