Getting Started with C# Programming

Jesse Liberty
XAMARIN MVP / MICROSOFT MVP

@jesseliberty http://jesseliberty.com

Variables

A variable is a value holder

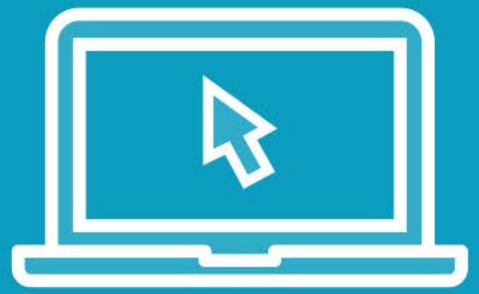
It has a type (e.g., int)

It has an identifier (e.g., myAge)

It has a value that may change during the running of the program

```
int myAge = 39;
```

Demo



Variables

if Statements

if (some expression) evaluates to true, do this action...

else: if the if statement is false, then do this...

Notes

= is used for assignmet: int x = 5;

== is used for equality: 5 + 3 == 8 // true

!= is used for inequality: 5 + 3 != 9 // true

```
Int x = 5;

Int y = 7;

if (x == 5 && y == 7){
    Console.WriteLine("this is true");
}

if (x == 5 || y == 9){
    Console.WriteLine("this is also true!");
}
```

Truth

&& is read “and,” – indicates that both parts must be true

|| is read “or,” – indicates that either part must be true

Truth Table

AND

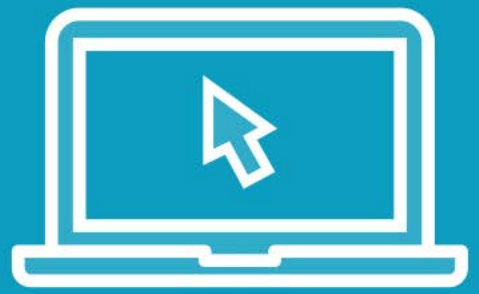
First	Second	Result
True	True	True
False	True	False
True	False	False
False	False	False

Truth Table

OR

First	Second	Result
True	True	True
False	True	True
True	False	True
False	False	False

Demo



if Statements

Incrementing

```
int x = 5;                  // x == 5
x = x + 1;                  // x == 6
x += 1;                     // x == 7
x++;                        // x = 8

int y = ++x;                // y == 9, x == 9
y = x++;                    // y == 9, x == 10
```

Decrementing

```
int x = 5;          // x == 5
x = x - 1;          // x == 4
x -= 1;             // x == 3
X--;                // x == 2

int y = --x;        // y == 1, x == 1
y = x--;            // y == 1, x == 0
```

For Loop

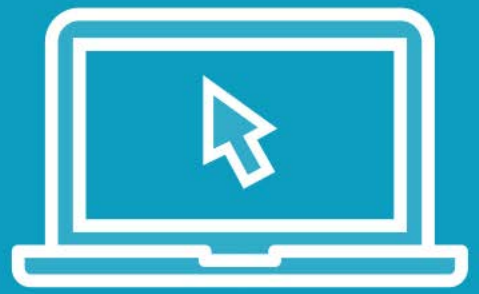
Simplest loop is for

for has three parts:

Initialization	for (**int i = 0**
Test	for (int i = 0; **i < 10**
Increment	for (int i = 0; i< 10; **i++**)

Beware of “off by one” or “fencepost” errors!

Demo



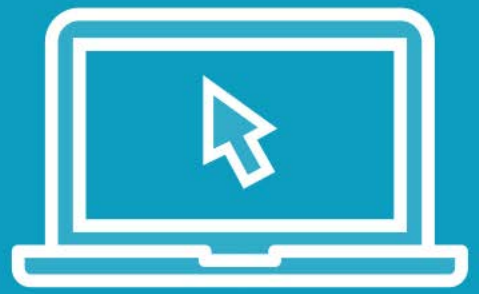
Looping with the for loop

While Loop and Do... While Loop

While some condition is true, do the following work
(while)

Do some work and then check if the condition is true
(do...while)

Demo



While Loop

Do ... While

Constants

Constants are assigned a value that never changes (hence the name!)

Can make your code more understandable

Makes it easier to fix a value used in many places

```
const double pi = 3.141592653589;
const int canDrinkAge = 21;
```

```
enum Actions{
    Create,
    Update,
    Delete,
}
```

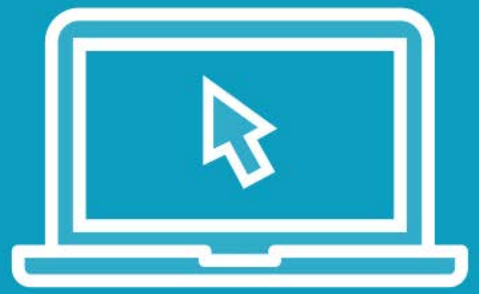
Enumerated Constants

Gives an individual value to each of a set

Can greatly simplify programming

Unless you tell it differently, starts at zero and increments by 1

Demo



Constants

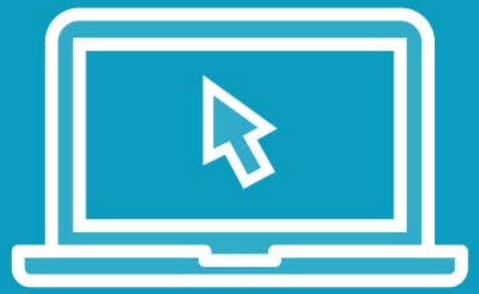
Enumerated Constants

Switch Statements

Much like if/else but much cleaner

Allows you to pick one path out of many, depending on a value

Demo



Switch

Summary

The meaning of truth

Looping and conditional statements

Enumerated constants for cleaner code