Swift 3 Fundamentals

A FEW MORE REASONS TO LEARN SWIFT

Simon Allardice
STAFF AUTHOR, PLURALSIGHT
@allardice www.pluralsight.com

Apple Development: **pre-Swift**

Client-side Development

Language:
Objective-C

Server-side Development

Node.js / JavaScript
RoR / Ruby
ASP.NET / C#
PHP
(etc.)

```
let toaster = Appliance();
toaster.model = "CrunchMaster 5000";
toaster.voltage = 120;
```

Semicolons aren't needed

```
no required import / include / using

no required main

print("Hello, Pluralsight")

no required return
```

A basic Swift program

```c
#include <stdio.h>

int main(void)
{
    [ your code goes here ]
    return 0;
}
```

A Simple Program **in C**

```
using System;
internal static class HelloWorld
{
    private static void Main()
    {
        [ your code goes here ]
    }
}
```

A Simple Program **in C#**

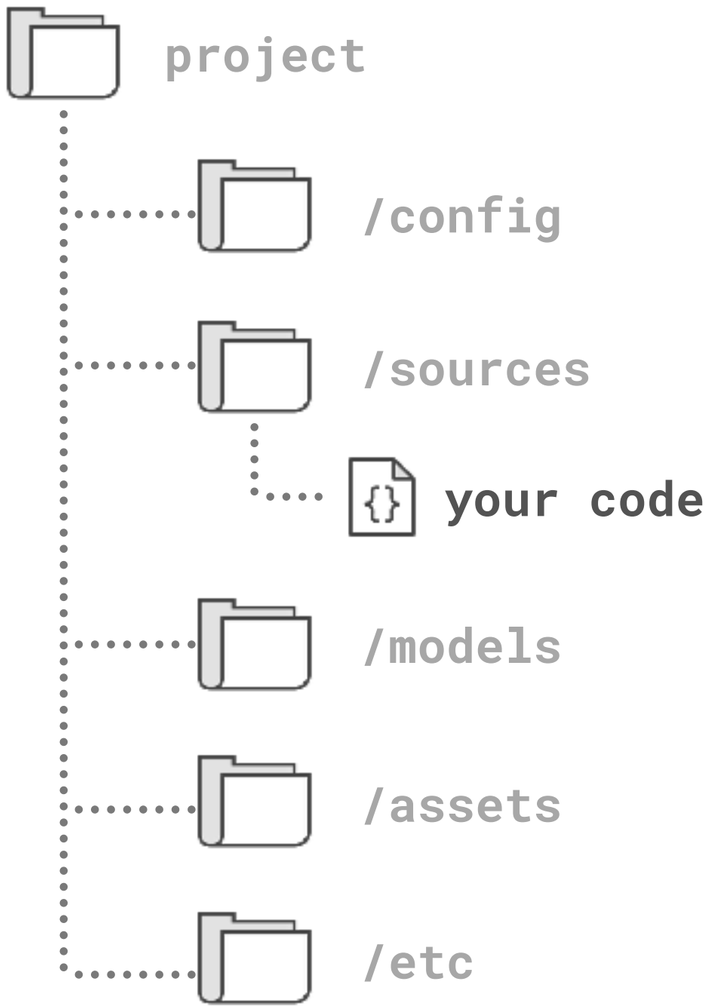
```
class HelloWorld {
    public static void main(String[] args) {
        [ your code goes here ]
    }
}
```

A Simple Program **in Java**

```
[ your code goes here ]
```

A Simple Program **in Swift**

Writing and Running Swift in a Playground


project
/config
/sources
your code
/models
/assets
/etc

```
name
firstName
dateOfBirth
jackOfAllTrades
```

Use Lower Camel Case for Variables

```
var playerName = "Alice"      // inferred as a Swift String
var age = 21                  // inferred as a Swift Int
var temperature = 72.6        // inferred as a Swift Double
var activeMember = true       // inferred as a Swift Bool
```

Type Inference

Swift *infers* the type from the initial value

```
var playerName = "Alice"
var age = 21
var temperature = 72.6
var activeMember = true
```

var is required

And is the *only* way to declare variables

SIDEBAR: The Swift Compilation Process

DEVELOPER

SHIP >

USER

SWIFT IS A COMPILED LANGUAGE

SOURCE CODE > FULL COMPILATION

MACHINE CODE

RUN

INTERMEDIATE (C#, JAVA)

SOURCE CODE > PARTIAL COMPILATION

BYTECODE

JIT COMPILE* > RUN

*VM / Runtime Engine required

INTERPRETED LANGUAGES

(JAVASCRIPT, RUBY)

SOURCE CODE - NO COMPILATION

SOURCE CODE

INTERPRET* > RUN

*Interpreter required