## Intro to TypeScript

In this lecture we will explain what TypeScript is and how we can code a web application using TypeScript.

**Learning Outcomes**

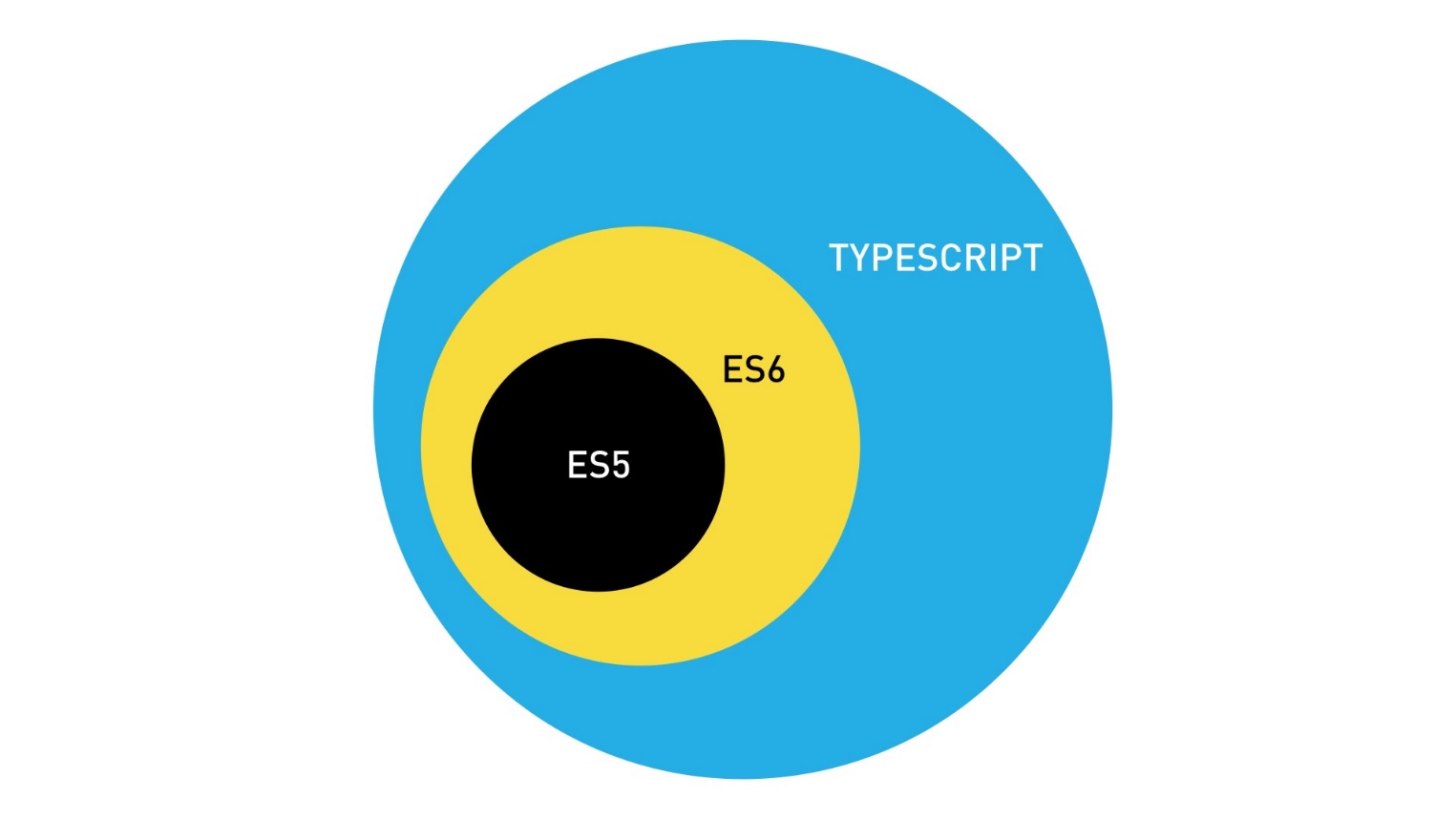
* Difference between TypeScript and JavaScript.
* How we convert from TypeScript into JavaScript.

**TypeScript vs JavaScript**

From the previous lecture we named the file we are going to type our code into script.ts

The reason the file ends in .ts instead of .js is that Angular is written in a superset of JavaScript called TypeScript.

TypeScript is the ES6 version of JavaScript plus a few other TypeScript only features which Angular needs in order to work.



You can write Angular applications in either *TypeScript*, *ES6* or even *ES5 JavaScript*.

However Angular itself is written in TypeScript, most examples on the web are written in TypeScript, most Angular jobs require you to write TypeScript so this book will be teaching in TypeScript.

**Transpilation**

Browsers don’t support TypeScript. Browsers *barely* support ES6 JavaScript. So how can we write our code in TypeScript?

We use something called a *transpiler* which converts from one language to another.

We can write in TypeScript and have a transpiler convert to ES6 or ES5.

 To understand why is it called *transpilation* and not *compilation*[/](https://www.stevefenton.co.uk/2012/11/compiling-vs-transpiling/) in more detail see [https://www.stevefenton.co.uk/2012/11/compiling-vs-transpiling](https://www.stevefenton.co.uk/2012/11/compiling-vs-transpiling/) (MORE INFO AT THE END OF THIS DOC.)

Since most browsers don’t support ES6 features yet we are going to *transpile* our TypeScript into ES5.

Later on in the book I’m going to show you how to transpile TypeScript into

JavaScript locally on your computer, but since we are using plunker we are using a feature of SystemJS which lets us transpile in the browser.

If we look at the tsconfig.json file in our demo plunker we can see there are a few settings we are using to convert TypeScript into JavaScript.

{

"compilerOptions": {

"target": "es5",

①

"module": "commonjs",

"moduleResolution": "node",

"sourceMap": true,

"emitDecoratorMetadata": true,

"experimentalDecorators": true,

"removeComments": false,

"noImplicitAny": true,

"suppressImplicitAnyIndexErrors": true

}

}

①

As you can see the target is set to ES5. We are telling the TypeScript transpiler to convert to ES5 javscript

**Summary**

* TypeScript is just JavaScript with a few more advanced features.
* Browser can’t run TypeScript so we first need to transpile it into JavaScript.
* The most common version of JavaScript is currently ES5 so we transpile TypeScript into ES5 JavaScript.

MORE INFO

[https://www.stevefenton.co.uk/2012/11/compiling-vs-transpiling](https://www.stevefenton.co.uk/2012/11/compiling-vs-transpiling/)

**Compiling**

is the general term for taking source code written in one language and transforming into another

**Transpiling**

is a specific term for taking source code written in one language and transforming into another language that has a similar level of abstraction

So (simplistically) when you compile C#, your method bodies are transformed by the compiler into IL. This cannot be called transpiling because the two languages are very different levels of abstraction.

When you compile TypeScript, it is transformed by the compiler into JavaScript. These are very similar levels of abstraction, so you could call this transpiling.