

Typescript – Syllabus

What is TypeScript?

TypeScript is a **programming language** developed and maintained by Microsoft. It is a **strict syntactical superset** of **JavaScript** and adds **optional static typing** to the language. It is **designed for the development** of **large applications** and **transpiles** to **JavaScript**.

Benefits:

Due to the **static typing**, code written in **TypeScript** is more **predictable**, and is generally **easier to debug**. **Google** is using **TypeScript** and **Angular** on **TypeScript** for its famous products such as **Google Analytics**, **Firebase**, and **Google Cloud Platform** including its **critical internal tools** - **bug tracking**, **employee reviews**, and product approval and launch tools.

Is good to opting a career in Typescript Scripting?

TypeScript demand is **growing** and is especially in demand among top tech companies where **pay** is **highest** **language** because it is popular or they desire the good features the said programming language has to offer. **TypeScript provides** highly **productive development tools** for **JavaScript IDEs** and practices, like **static checking**. **TypeScript** makes code **easier** to **read** and **understand**.

What is the scope of Typescript Scripting?

Typescript is **open-source** so there is a **future**. **Typescript** is often promoted as **superset of Javascript**. It has **features** of a **statically typed language**. We all know **JS** is **dynamic** in **nature** and **working** on **large application** can **become difficult**. **Typescript** helps to **organize** and **maintain large codebase** offering some **features** of **typed language** like **interface**, **classes** etc. As you can **develop front-end** in **major technologies** using **typescript**.

Is Typescript Scripting will enough to create a job?

Yes, It has a **great scope** in **job market**. **Typescript** has gained popularity in a short span of time, where typescript has a great future. Since **TypeScript works** with **JavaScript** it can be **used** for either writing both **frontend** or **backend**, but still **compiles** to **JavaScript**. **Highest-paid programming** and **scripting languages**.

What you learn in Typescript Scripting?

In this course, you'll start from the beginning and work your way up to more advanced concepts like **type checking**, **iterators**, and **manipulating objects** and **arrays**. You'll get a chance to get your hands dirty with plenty of real practice along the way. TypeScript addresses many of the pain points in JavaScript, while "sitting on top of" it - everything possible in JavaScript is also available in TypeScript.

Pre-requisites to learn Typescript Scripting?

There are no specific prerequisites to learn TypeScript Scripting. Having Basic knowledge of the Windows Operating System is recommended.

Is Typescript the future?

The **Future of TypeScript is Bright**. Today, **.NET Framework** has become an **enterprise name** to **build Windows, Web, and mobile applications** and **C#** is the **prime language** to build. **TypeScript** provides **highly productive development tools** for **JavaScript IDEs** and practices, like **static checking**. **TypeScript** makes **code easier** to **read** and **understand**.

Why we are using Typescript in Angular?

Angular is **built** using **TypeScript** which brings **many benefits** to the table such as: **TypeScript** is a **superset of JavaScript**. **TypeScript** is not its **own stand-alone language** like **CoffeeScript**, **Dart** or others and that's super powerful. **TypeScript** provides **support** for **types** (**primitives**, **interfaces**, and other **custom types**). With **TypeScript**, we can make a **huge improvement** over **plain JavaScript**. **TypeScript** gives us all the **benefits of ES6** (ECMAScript 6), plus **more productivity**.

Top reasons to consider a career in Typescript Scripting?

TypeScript demand is **growing** and **development** of **Angular applications** involves **writing** the **code** using **Typescript**, a **superset of Javascript**, along with HTML. Due to the **static typing**, code written in **TypeScript** is **more predictable**, and is generally **easier to debug**. **Career** and **Salary** as an **Angular developer**, a **lot of scope** for **progressive growth** at the **beginning** of the **career**. learn **TypeScript** to be a **good developer**, most people do just fine without it. However, working with **TypeScript** definitely has its **benefits**: Due to the **static typing**, **code written** in **TypeScript** is **more predictable**.

| Sl. no | Topic | Hrs |
|----------|--|-----|
| | | |
| 1 | Typescript Introduction | |
| | Typescript Introduction | |
| | Advantages or Features of Typescript | |
| | Differences between Typescript & Javascript | |
| | Configure or Setup of Typescript Environment | |
| | Write a First Program in Typescript | |
| | Creating a Project using Typescript | |
| 2 | ES6 Language Features | |
| | What is ES6? & Language Features | |
| | Default Parameters | |
| | Template String | |
| | Let and Const | |
| | For...of Loops | |
| | Lambdas | |
| | Destructuring | |
| | The Spread Operators | |
| | Computed Properties | |
| 3 | Type Fundamentals | |
| | Understanding Javascript Types | |
| | Understanding type interface | |
| | Specifying Javascript Types | |
| | Specifying function parameter types | |
| | Adding function overloads | |
| 4 | Custom Types | |
| | Define custom types with interfaces | |

| | | |
|----------|--|--|
| | Using interfaces to describe functions | |
| | Extending interface definitions | |
| | Defining constant values with enums | |
| | Defining anonymous types | |
| 5 | Classes | |
| | Understanding prototypical inheritance | |
| | Defining a class | |
| | Applying static properties | |
| | Making properties smarter with accessors | |
| | Inheriting behavior from a base class | |
| | Implementing an abstract class | |
| | Controlling visibility with access modifiers | |
| | Implementing interfaces | |
| 6 | Generics | |
| | Introducing Generics | |
| | Creating generic classes | |
| | Applying generic constraints | |
| 7 | Modules | |
| | Understanding the need for Modules in Javascript | |
| | Organizing your code with namespaces | |
| | Using namespaces to encapsulate private members | |
| | Understanding the difference between Internal and External Modules | |
| | Switching from Internal to External Modules | |
| | Importing modules using CommonJS syntax | |
| | Importing modules using ECMAScript2015 syntax | |
| | Loading External modules | |

| | | |
|----------|--|--|
| 8 | Real-World Application Development | |
| | Introducing sample Javascript Application | |
| | Converting existing Javascript code to Typescript code | |
| | Generating declaration files | |
| | Refencing 3 rd party libraries | |
| | Debugging Typescript with source maps | |
| 9 | Decorators | |
| | Implementing method decorators | |
| | Implementing class decorators | |
| | Implementing property decorators | |
| | Implementing decorator factories | |



T S