LAB 1: Getting started with Typescript

In this first lab, you will setup the development environment, install Typecript and compile your first Typescript file.

Install requirements

1. Download and install NodeJs: https://nodejs.org/en/
2. Download and install VSCode: https://code.visualstudio.com/download
3. Run the following command to install **typescript** into your machine:

```
npm install -g typescript
```

> In some Windows editions, running scripts is disabled, to enable them run **Powershell as Administrator** and exceute the following command: `Set-ExecutionPolicy -ExecutionPolicy Unrestricted`

4. Run the following command to check of the TS compiler `tsc` is correctly installed

```
tsc -v
```

Create first project

1. Create an empty folder: `Lab1` and open it with VSCode
2. Initialize the folder as a Typesript project using the following command:

```
tsc --init
```

> A file `tsconfig.json` is created which defines Typescript compiler options

3. Create a new file `hello.ts`
4. Copy the following code snippet, it should print a hello world message

```
console.log("Hello world");
```

Compile typescript

1. Invove the typescript compiler to compile typescript files (`.ts`) into javascript files (`.js`)

```
tsc
```

> After runnning `tsc` , for each TS file a correponsing JS file is generated, by default the output version (ECMAScript edition) is **ES2016** acording to `tsconfig.json`

2. Now, we can include javascript files into a web project, or run them direclty using NodeJS runtime, for example:

```
node hello.js
```

> This should print a message "**hello world**" in the console

Customize the compilation

We can customize and change the behavior of the Typescript compiler by changing the options in the `tsconfig.json` or provinding options for the `tsc` command.

1. By default, `tsc` command compiles every file found in the root directry (at the tsconfig level), to specify the root folder within your source files, we can change the value `rootDir` in `tsconfig.json` , for example:

```
"rootDir": "./src",
```

> By providing this option, this tells the compiler what TS files to include in the compilation

2. By default the output folder of the compilation is the project root directory itself, we can change it by setting the value `outDir` in `tsconfig.json` , for example:

```
"outDir": "./dist",
```

> By providing this option, this tells the compiler to place the output files (.js files) into a `dist`

> folder instead of the root directory