2 - Initialize Video Processing Service

10:56

Initialize Video Processing Service (Express TypeScript Boilerplate)

> You can find the finished code for this course **here** for your reference. Most lessons have a commit

> associated with them. You can find the commit history via `git log`.

We will be using Express.js and TypeScript for our Video Processing Service.

There are ways we can auto-generate the boilerplate code for Express.js and TypeScript. But we will do so


```
mkdir video-processing-service
cd video-processing-service
npm init -y
```

This will create a new `package.json` file with default values.

2. Install Express, TypeScript, and the TypeScript Node development server as dependencies:

```
npm install express
npm install --save-dev typescript ts-node
```

3. Install the type definitions for Node.js and Express:

```
npm install --save-dev @types/node
@types/express
```

TypeScript will automatically use @types packages to provide type definitions for Node.js and Express.

Basically, TypeScript code goes through a *build step* where it is converted into JavaScript. The type definitions are used during this build step to ensure that your code is type-safe (no mismatching types). If there are errors, the build will fail.

4. Create a `tsconfig.json` file for TypeScript configuration:

```
{
  "compilerOptions": {
    "target": "es6",
    "module": "commonjs",
    "rootDir": "src",
    "outDir": "dist",
    "strict": true,
    "esModuleInterop": true},
  "include": ["src/**/*.ts"],
  "exclude": ["node_modules"]
}
```

5. Update your `package.json` scripts:

```
"scripts": {
  "start": "ts-node src/index.ts",
  "build": "tsc",
  "serve": "node dist/index.js"
}
```

6. Create a new `src` directory with an `index.ts` file for your Express app:

```
import express from 'express';

const app = express();
const port = 3000;
```

```
app.get('/', (req, res) => {
  res.send('Hello World!');
});

app.listen(port, () => {
  console.log(`Server running at
http://localhost:${port}`);
});
```

7. Now you can start your Express.js server with `npm run start`.