TypeScript

Modules

Content

1. Introduction
2. Creating a new module
3. Export statements
4. Importing a new module
5. Default Exports

1. Introduction

- A module executes within its own scope, not in the global scope. It means that when you declare variables, functions, classes, interfaces, etc., in a module, they are not visible outside the module unless you explicitly export them using export statement.
- On the other hand, if you want to access variables, functions, classes, etc., from a module, you need to import them using the import statement.

2. Creating a new module

- The following creates a new module called Validator.ts and declares an interface named Validator

```
export interface Validator {
    isValid(s: string): boolean
}
```

3. Export statements

```
interface Validator {
    isValid(s: string): boolean
}

export { Validator };
```

- Or you can rename declarations for module consumers

```
interface Validator {
    isValid(s: string): boolean
}

export { Validator as StringValidator };
```

4. Importing a new module

```
import { Validator } from './Validator';

class EmailValidator implements Validator {
    isValid(s: string): boolean {
        const emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;
        return emailRegex.test(s);
    }
}

export { EmailValidator };
```

- You can rename when import

```
import { Validator as StringValidator } from './Validator';

class EmailValidator implements StringValidator {
    isValid(s: string): boolean {
        const emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;
        return emailRegex.test(s);
    }
}

export { EmailValidator };
```

5. Default Exports

- TypeScript allows each module to have one default export. Use the default keyword to mark default

```
import { Validator } from './Validator';

export default class ZipCodeValidator implements Validator {
    isValid(s: string): boolean {
        const numberRegexp = /^[0-9]+$/;
        return s.length === 5 && numberRegexp.test(s);
    }
}
```

```
import ZipCodeValidator from './ZipCodeValidator';

let validator = new ZipCodeValidator();
let result = validator.isValid('95134');

console.log(result);
```

THE END