



TypeScript Moduly

kurz Vývoj progresívnych
webových aplikácií

Eduard Kuric

Moduly

- Sú súčasťou JavaScriptu od ES6
- Môže obsahovať deklarácie aj kód
- Má vlastný rozsah platnosti
 - Deklarované premenné, funkcie, triedy, rozhrania v module majú platnosť v module, a teda nie sú viditeľné mimo modlu
- Ak chceme - premenné, funkcie, triedy, atď. - sprístupniť „vonkajšiemu svetu“, použijeme **export** statement
- Na strane druhej, ak chceme použiť/pristúpiť k
 - premenným, funkciám, triedam modulu - použijeme **import** statement

Modul `validator.ts`

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

- bez `export` by bolo rozhranie `Validator` v režime `private`, nebolo by možné použiť ho okolitým svetom (napr. iným modulom)

export

- Iný spôsob exportu

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

```
export { Validator };
```

export – as

- Premenovanie deklarácie rozhrania

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

- Iné moduly budú používať StringValidator interface, namiesto Validator interface

import

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

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

// export class
export { EmailValidator };
```

import – as

- Premenovanie rozhrania

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

app.ts — použitie validátora

```
import { EmailValidator } from './EmailValidator';  
  
let email = 'john.doe@typescripttutorial.net';  
let validator = new EmailValidator();  
let result = validator.isValid(email);
```


Importovanie typov – types.ts

```
export type alphanumeric = string | number;
```

```
import type {alphanumeric} from './Types';
```

Importovanie všetkého *

- použijeme hviezdičku * :

```
import * from 'module_name';
```

Re-export

- Majme `EmailValidator.ts` a `ZipCodeValidator.ts`
- Vytvorme nový modul `FormValidator.ts`, ktorý skombinuje exporty oboch validátorov

```
// FormValidator.ts  
export * from "./EmailValidator";  
export * from "./ZipCodeValidator";
```

export default

```
// main.ts
class Users {
    fetchUsers() {}
}
class Posts {
    fetchPosts() {}
}
// vieme exportovat pomenovane deklaracie
export {Users, Posts}
-----
import {Users, Posts} from `./main`;
```

export default //2

```
// users.ts
```

```
// nepotrebuje explicitny nazov deklaracie
```

```
// nazvy su optional
```

```
export default class {  
    fetchUsers() {}  
}
```

```
-----
```

```
// mozeme tym padom pouzit lubovolny nazov
```

```
import UsersFactory from "./Users";
```

export default //3

```
// main.ts
```

```
class Users {  
    fetchUsers() {}  
}
```

```
export default class Posts {  
    fetchPosts() {}  
}
```

```
export {Users};
```

```
-----
```

```
import {Users}, Posts from "./main.ts";
```

export default //3

```
// main.ts
```

```
class
```

```
  f
```

```
}
```

```
export
```

```
  f
```

```
}
```

```
export
```

*v module může být iba jeden
export default*

```
-----
```

```
import {Users}, Posts from "./main.ts";
```