

MỤC TIÊU:

Kết thúc bài thực hành này bạn có khả năng

- ✓ Sử dụng arrow function, function với tham số.
- ✓ Kết hợp DOM, arrow function để hiển thị đối tượng, mảng đối tượng trong typescript lên HTML.
- ✓ Biên dịch nhiều tập tin typescript

NỘI DUNG

Bài 1: arrow function

1. Viết hàm (function và arrow function) tính tổng có giá trị trả về không dùng tham số.
2. Viết hàm arrow function tính tổng 2 số (có sử dụng default value, optional parameter, rest parameter)
3. Thực hiện ví dụ merging array with spread operator

```
TS app.ts  ×
SimpleProject > TS app.ts > ...
1  //ARRAY
2  const hobbies = ['Sports', 'Cooking'];
3  const activehobbies = ['Hiking'];
4  activehobbies.push(hobbies);
5  activehobbies.push(hobbies[0], hobbies[1]);
6  activehobbies.push(...hobbies);
7  console.log(activehobbies);
8
9
10
11
12
```

Bài 2

Thực hiện bài tập ví dụ

1. Function & void

```
TS app.ts ×
SimpleProject > TS app.ts > [?] sum
1  let sum = (x:number=5, y?:number) => { return x + <number>y; }
2  let speech = (output: any): void => {
3  |    console.log("Result:" + output);
4  |  }
5  speech(sum(5,12));
6  console.log(speech(sum(8,5)));
7
8
9
```

2. Never & void

```
TS app.ts ×
SimpleProject > TS app.ts > ...
1  let something: void = undefined;
2  let nothing: never = null; // Error: Type 'null' is not assignable to type 'never'
3  function throwError(errorMsg: string): never {
4  |    throw new Error(errorMsg);
5  |  }
6
7
8
9
```

3. Function & callback

```
TS app.ts ×
SimpleProject > TS app.ts > ...
1  function AddandHandle(x: number, y: number, cb:(num:number)=>void){
2  |    const result = x + y;
3  |    cb(result);
4  |  }
5  AddandHandle(10, 20, (result) => { console.log(result); })
6
7
8
9
```

Bài 3: Project LEARNING TYPESCRIPT

Tạo simple project có cấu trúc sau:

- Tên project: LEARNING_TYPESCRIPT
- Project gồm: src/app.ts (biên dịch đến dist/app.js), index.html, style.css, tsconfig.json

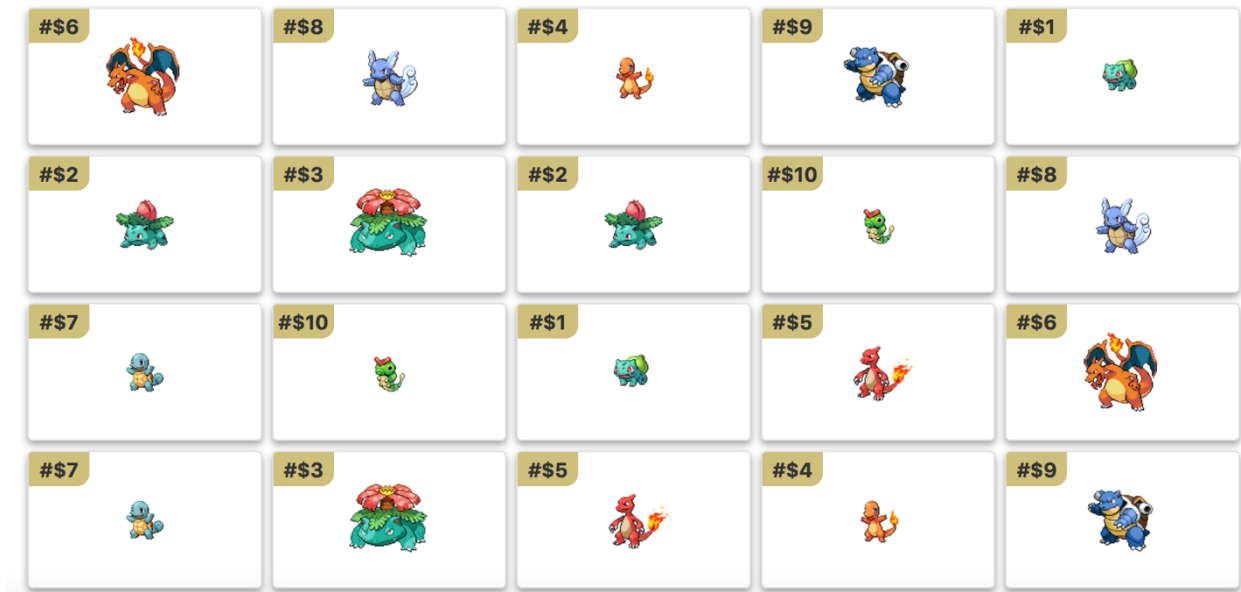
Index.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link rel="stylesheet" href="public/style.css" />

    <title>Learning TypeScript</title>
  </head>
  <body>
    <main>
      <h1>Typed Pokedex</h1>
      <div id="app">
      </div>
    </main>
    <script src="public/js/app.js"></script>
  </body>
</html>
```

app.ts

- Tạo đối tượng với các thuộc tính
 - id: number;
 - name: string;
 - image: string;
 - type: string;
- Lấy n (10-20) đối tượng từ đường dẫn sau (https://pokeapi.co/api/v2/pokemon) và hiển thị ở trang index.html
 - Xáo trộn (**arrow function shuffle**) các đối tượng và sử dụng DOM để hiển thị (**arrow function show**).
 - Hình mẫu



Gợi ý:

```
data = shuffle (data);
show(data);
```

```
const shuffle = (data: []): [] => {
  return data;
}
```

```
const show = (data: []): void => {
  //code here use DOM
}
```

Bài 4

Giảng viên cho thêm

*** Yêu cầu nộp bài:

SV nén file (hoặc share thư mục google drive) bao gồm các yêu cầu đã thực hiện trên, nộp LMS đúng thời gian quy định của giảng viên. KHÔNG NỘP BÀI COI NHƯ KHÔNG CÓ ĐIỂM.

--- Hết ---

