

MỤC TIÊU:

Kết thúc bài thực hành này bạn có khả năng

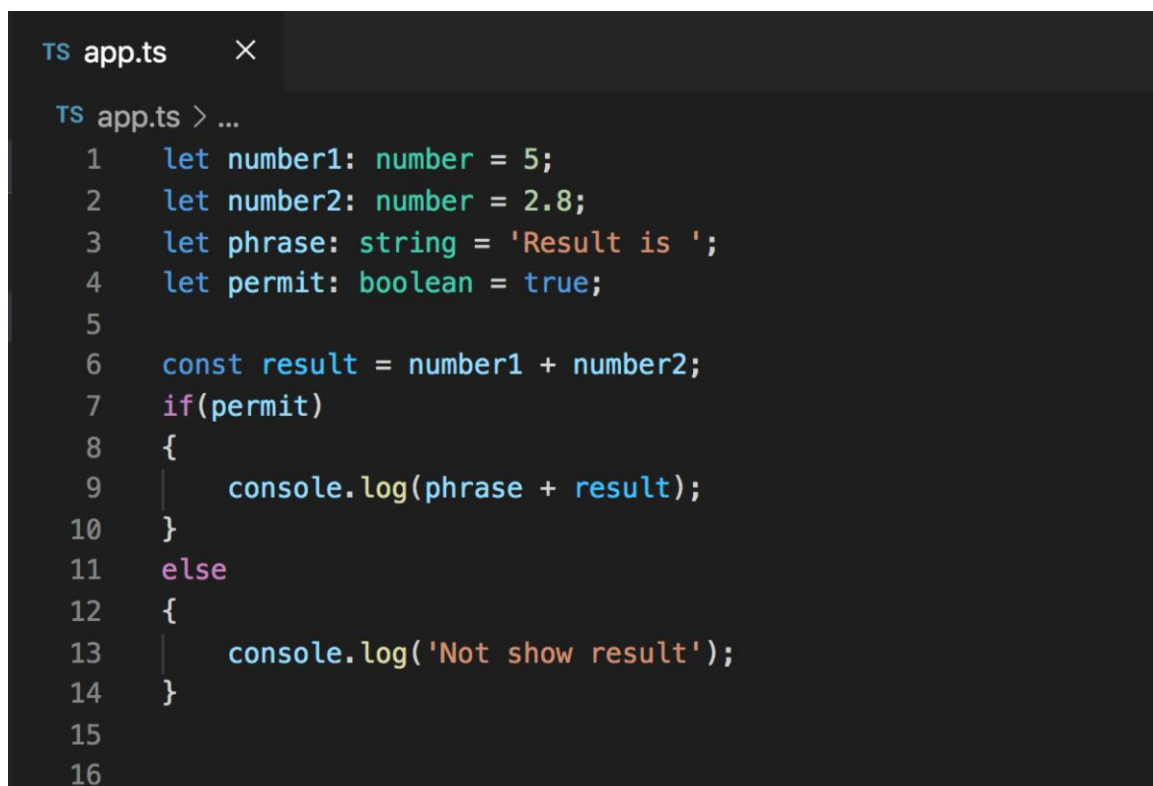
- ✓ Sử dụng các loại dữ liệu cơ bản trong typescript.
- ✓ Sử dụng các loại dữ liệu thêm mới bởi typescript.
- ✓ Kết hợp DOM để hiển thị đối tượng, mảng đối tượng trong typescript lên HTML.

NỘI DUNG

Bài 1

Thực hiện các bài tập ví dụ.

1. Sử dụng number, string, boolean



```
TS app.ts  ×
TS app.ts > ...
1  let number1: number = 5;
2  let number2: number = 2.8;
3  let phrase: string = 'Result is ';
4  let permit: boolean = true;
5
6  const result = number1 + number2;
7  if(permit)
8  {
9      console.log(phrase + result);
10 }
11 else
12 {
13     console.log('Not show result');
14 }
15
16
```

2. Type inference

```
TS app.ts ×
TS app.ts > [🔍] result
1  function add( x = 5){
2
3      let phrase = 'Result is ';
4
5      phrase = 10;
6      x = '2.8';
7
8      return phrase + x;
9  }
10
11 let result: number = add();
```

3. Object

```
TS app.ts ×
TS app.ts > ...
1  var person : {
2      name: string,
3      age: number
4  }
5
6  person = {
7      name: "Typescript",
8      age: 11
9  }
10
11 console.log(person.name);
12
13
```

Bài 4

Thực hiện bài tập ví dụ

1. Array, tuple, any, enum

```
TS app.ts  ×
TS app.ts > ...
1  enum Role {ADMIN, READ_ONLY, AUTHOR};
2  const person : {
3      name: string,
4      age: number,
5      hobbies: string[],
6      role: string,
7      roletuple: [number, string]
8  } = {
9      name: 'Typescript',
10     age: 11,
11     hobbies: ['Sports','Cooking'],
12     role: Role.ADMIN, //Error
13     roletuple: [2, 'author']
14 }
15
16 let favouriteActivites: any[];
17 favouriteActivites = [5, 'Sports', true];
18
19 if(person.role === Role.AUTHOR){
20     console.log('is author');
21 }
22
23 person.roletuple.push('admin');
24 person.roletuple[1] = 10; //Error
25 person.roletuple = [0, 'admin', 'user']; //Error
26
27
28
```

2. Literal type & custom type

```
TS app.ts  X
TS app.ts > ...
1  type Combinable = number | string;
2  function combine(input1: Combinable, input2: number|string, resultConversion: 'as-number' | 'as-text'){
3      let result;
4      if(typeof input1==='number' && typeof input2==='number' || resultConversion==='as-number'){
5          result = parseFloat(input1) + parseFloat(input2);
6      } else { //concatenated
7          result = input1.toString() + input2.toString();
8      }
9      return result;
10 }
11
12 const combineNumber = combine(30, 26, 'as-number');
13 console.log(combineNumber);
14
15 const combineStringNumber = combine('30', '26', 'as-number');
16 console.log(combineStringNumber);
17
18 const combineText = combine('Typescript Vs ', 'Javascript', 'as-text');
19 console.log(combineText);
20
21
22
23
24
25
26
```

3. Num & undefined

```
TS app.ts  X
TS app.ts > ...
1  //Variable declared and assigned to null
2  var a = null;
3  console.log( a );
4  console.log( typeof(a) );
5
6  //Variable declaration without assigning any value to it
7  var b;
8  console.log( b );
9  console.log(typeof(a));
10 console.log(undeclaredVar);
```

4. Unknown & any

```
TS app.ts ×
TS app.ts > ...
1 let userInput: unknown;
2 let userName: string;
3
4 userInput = 5;
5 userInput = 'Typescript';
6 userName = userInput;
7 userName = <string> userInput;
8 if(typeof userInput === 'string'){
9     userName = userInput;
10 }
```

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

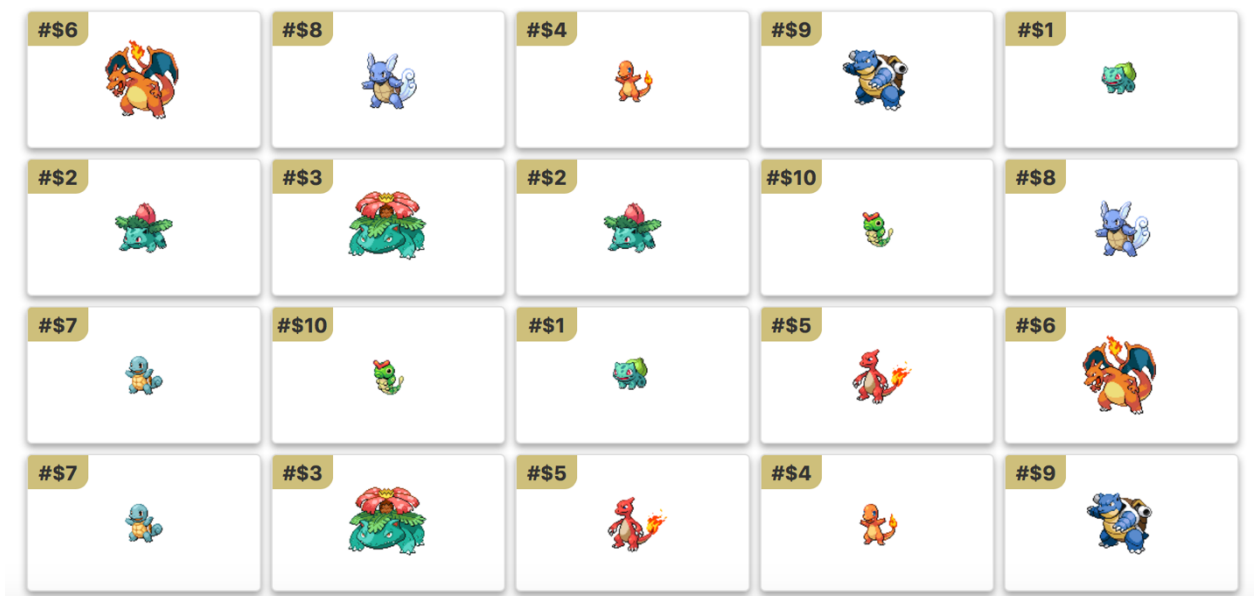
Index.html

```
<!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 (***function shuffle***) các đối tượng và sử dụng DOM để hiển thị.
 - Hình mẫu



Gợi ý:

```
const pokemons: number = 10
//i:1...10
let data: Response = await fetch(`https://pokeapi.co/api/v2/pokemon/${i}`)
let pokemon: any = await data.json()
```

Bài 5

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 ---