# A Short Typescript(Javascript) top-up

# TS Features

The following features are used in the labs/examples

- Destructuring
- Spread Operator
- Default Arguments
- Array methods

The active selection you've provided, ${v1}, is a template literal in JavaScript and TypeScript

# Destructuring

- Assigning the properties of an array or object to variables using a declarative style rather than an imperative/procedural style.

```
// ARRAYS
{
    // without destruction
    const nums = [10, 11, 12]
    let v1 = nums[0]
    let v2 = nums[1]
    let v3 = nums[2]
}
```

- See 01_destructing.ts

```
{
    // with destructuring
    const nums = [10, 11, 12]
    let [v1, v2, v3] = nums
    console.log(`${v1} ${v2} ${v3} `)  // 10 11 12
}
```

Template Literal. A way of output variables in the string

# The Spread Operator

- JS arrays and objects are iterables.
- The spread operator allows an iterable to expand in places where 0+ arguments are expected.
- Syntax:

```
…arrayRef
…objectRef
```

# Default arguments

```
function greet(name: string = 'Guest'): string {
  return `Hello, ${name}!`;
}

console.log(greet()); // "Hello, Guest!"
console.log(greet('John')); // "Hello, John!"
```

# Array Methods

- find(...) returns the value of the first element in the array where predicate is true, undefined otherwise.

```
// Use the find method to find the first user whose age is great
let user = users.find((user: {age:number })=> user.age > 30);
console.log(user); // Output: { id: 3, name: 'Doe', age: 35 }
```

- map(...) executes a callback function on each element of an array, and returns an array that contains the results.

```
// Use the map method to create a new array containing the names of all users
let names = users.map(user => user.name);
console.log(names); // Output: ['John', 'Jane', 'Doe']
```

- Filter(...) returns the elements of an array that meet the condition specified in a callback function.

```
// Use the filter method to create a new array containi
let olderUsers = users.filter(user => user.age > 25);
console.log(olderUsers); // Output: [{ id: 2, name: 'Ja
```