

Generics

AGENDA

1. Generics
 2. keyof operator
 3. typeof operator
 4. Mapped types
- Hiểu nôm na: kiểu dữ liệu mà có nhận tham số và trả về kiểu dữ liệu tương ứng.

```
interface Student {  
  id: number;  
  name: string;  
}  
  
const numberList: Array<number> = [1, 2, 3];  
const wordList: Array<string> = ['easy', 'frontend'];  
  
const studentList: Array<Student> = [  
  { id: 1, name: 'Alice' },  
  { id: 2, name: 'Bob' },  
]
```

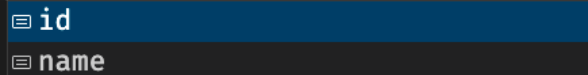
Thử định nghĩa kiểu generic tương tự như Array:

- Index signature là kiểu **string** --> các props khác phải compatible với type của index signature.

```
interface Student {  
  id: number;  
  name: string;  
}  
  
interface List<T> {  
  length: number;  
  [index: number]: T;  
}  
  
const numberList: List<number> = [1, 2, 3];  
const wordList: List<string> = ['easy', 'frontend'];  
  
const studentList: List<Student> = [  
  { id: 1, name: 'easy' },  
  { id: 2, name: 'frontend' },  
]
```

keyof operator

```
1 interface Student {
2   id: number;
3   name: string;
4 }
5
6 type StudentKeys = keyof Student;
7
8 const keys: StudentKeys = ''
```



typeof operator

```
console.log(typeof 'easy frontend'); // 'string'
console.log(typeof 123); // 'number'
console.log(typeof false); // 'boolean'
console.log(typeof {}); // 'object'
console.log(typeof []); // 'object'
console.log(typeof function() {}); // 'function'
console.log(typeof window !== 'undefined'); // true on client, false on server
```

Mapped types

- Dùng index signature để định nghĩa kiểu dữ liệu có props lấy từ kiểu dữ liệu khác.

```
interface Student {
  id: number;
  name: string;
}

type MappedTypes = {
  [Key in keyof Student]: boolean;
};
```

Tham khảo: <https://www.typescriptlang.org/docs/handbook/2/mapped-types.html>

Tham khảo

- <https://www.typescriptlang.org/docs/handbook/2/types-from-types.html>
-

Series - Typescript cơ bản 🎉

- Tác giả: **Hậu Nguyễn**
- Được phát hành trên kênh youtube **Easy Frontend**.
- Tài liệu pdf và videos đều có bản quyền thuộc về Easy Frontend.
- Videos được phát hành cho fan cứng trước, public sau.
- [Đăng ký fan cứng](#) để xem series này đầy đủ và sớm nhất nhé.

Kết nối với mình

-  Follow Facebook: <https://www.facebook.com/nvhauesmn/>
-  Like Fanpage: <https://www.facebook.com/learn.easyfrontend>
-  Youtube Channel: <https://www.youtube.com/easyfrontend>