

MỤC TIÊU:

Kết thúc bài thực hành này bạn có khả năng

- ✓ Sử dụng interfaces.
- ✓ Sử dụng class, abstract class.
- ✓ Kế thừa (inheritance), access modifiers.

NỘI DUNG

Bài 1: Interface như là 1 function

Thực hiện bài tập ví dụ

```
interface AddFn {  
  (a: number, b: number): number;  
}  
  
let add: AddFn;  
  
add = (n1: number, n2: number) => {  
  return n1 + n2;  
};
```

Bài 2: Interface và kế thừa

Thực hiện bài tập ví dụ

```
interface Named {  
  readonly name?: string;  
  outputName?: string;  
}  
  
interface Greetable extends Named {  
  greet(phrase: string): void;  
}
```

```
let user1: Greetable;

user1 = new Person();
// user1.name = 'Manu';

user1.greet('Hi there - I am');
console.log(user1);
```

Bài 3: Class

Thực hiện bài tập ví dụ

```
abstract class Department {
    static fiscalYear = 2020;

    protected employees: string[] = [];

    constructor(protected readonly id: string, public name: string) {
    }

    static createEmployee(name: string) {
        return { name: name };
    }

    abstract describe(this: Department): void;

    addEmployee(employee: string) {
        this.employees.push(employee);
    }

    printEmployeeInformation() {
        console.log(this.employees.length);
        console.log(this.employees);
    }
}
```

```
class ITDepartment extends Department {
  admins: string[];
  constructor(id: string, admins: string[]) {
    super(id, 'IT');
    this.admins = admins;
  }

  describe() {
    console.log('IT Department - ID: ' + this.id);
  }
}

const employee1 = Department.createEmployee('Max');
console.log(employee1, Department.fiscalYear);

const it = new ITDepartment('d1', ['Max']);

it.addEmployee('Max');
it.addEmployee('Manu');
```

Bài 4:

- Tạo các class liên quan đến cơ sở dữ liệu của Asm
- Hoàn thiện web user và admin dùng class

*** 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 ---