Npm install –g @angular/cli

Ng new my-app

Cd my-app

Ng serve –open ⬄ ng s --open

Ng generate component **hero-detail** ⬄ ng g component hero-detail

(hero-detail.component.ts, export class **HeroDetailComponent**{…} )

Ng generate **service hero** ⬄ ng g service …

(hero.service.ts, export class **HeroService** {…})

Ng generate module app-routing –flat –module=app

--flat puts the file in src/app instead of its own folder.  
--module=app tells the CLI to register it in the imports array of the AppModule.

(module cần thêm –module để nó thêm vào)

HttpClient.get() returns an Observable

Observable is one of the key classes in the [RxJS library](http://reactivex.io/rxjs/).

getHeroes(): Observable<Hero[]> { return of(HEROES); }

of(HEROES) returns an Observable<Hero[]> that emits a single value, the array of mock heroes.

{{biến}}

[thuộc tính]

(event)

[ngModel] : one-way binding, cần FormsModule

[(ngModel)] : two-way binding, cũng cần FormsModule

<input #username placeholder=”username”/> : #username là id trong angular để gọi đến, đại diện cho thẻ input, để lấy dữ liệu input khi được nhập vào

( <button (click)=”getUsername(username.value)”> Add </button> )

Input decorator :

@Input() hero: Hero; (hero-detail.component.ts)

<hero-detail [hero]=”selectedHero”></hero-detail> (hero-detail.html)

Sử dụng template literal:

const apiUrl = ‘example.com’;

id = 1;

const url = ` ${this.apiUrl}/${this.id} `;

Async Pipe : <https://tinyurl.com/y8fqvcng>

Các pipe: <p>{{text **| upper**}}

Export class Employee {

Constructor() {

Public id: number;

…

**Public jobCategory?: string;** (có nghĩa là có thể có giá trị hoặc null)

}

}

In this case, the ?: is not a single operator, but rather two operators:

* ? (optional),
* : (specify type).