**Câu hỏi audit module 5**

1. Trình bày được ý nghĩa và khái niệm Observer

* Là một object chứa các phương thức next , error , complete để xử lý dữ liệu tương ứng được gửi từ observable
* Là subcribe observable . Về cơ bản observer là những callback cho các sự kiện mà giá trị được gửi đi từ observable

1. Trình bày được ý nghĩa và khái niệm Observable

* Là thành phần xử lí bất đồng bộ ở angular rxjs.

1. Phân biệt sự khác nhau giữa var, let, const

* Const là hằng số . không thể thay đổi giá trị.
* Var có thể tái tạo lại , cơ chế hoisting , function scope
* Let chỉ có thể thay đổi giá trị của biến mà không thể tạo lại , block scope

1. File angular.json và package.js sử dụng để làm gì?

* Angular.json dùng để cấu hình lại angular CLI
* Package.js file này chứa các thư viện thứ 3 cần thiết cho dự án , ngoài ra chúng ta muốn thêm thư viện thứ 3 thì khai báo trong này

1. Kể tên các directive được hỗ trợ trong angular

* Component directive
* Stuctural directive : ngFor ngIF ngSwitch
* Attribute directive: ngstyle ngclass

1. Trình bày được ý nghĩa và khái niệm subcribe

* Khi làm việc với observable thì thường xuyên sử dụng subscribe để theo dõi các thay đổi của dữ liệu hoặc nhận về kết quả của 1 công việc nào đó .

1. Trình bày khái niệm Single page application

* Là ứng dụng web 1 trang , giúp nâng cao trải nghiệm người dùng.Và việc hoán đổi từng trang đã có router đảm nhận nhiệm vụ này.

1. Trình bày về Data binding trong component

* Có 2 loại chính là one way binding và two way binding
* One way binding : gồm có interpolation , property binding , eventbinding.
* Two way binding : [(ngModel)]

1. Trình bày sự khác nhau giữa navigateByUrl và navigate

* Navigate : Truyền 1 array gồm các đoạn của url để chuyển từ view này sang vỉew khác
* Navigate: Truyền 1 String.

1. TypeScript là gì?

* Là superset của Javasript
* Thiên về OOP và có kiểu dữ liệu rõ ràng.
* Được dịch thành Js để chạy trên trình duyệt

1. Tại sao nên sử dụng Typescript, Ưu điểm của Typescript

* Vì hỗ trợ tổ chức code rõ ràng hơn , hỗ trợ cơ chế hệ thống code theo từng module giúp lập trình viên có thể làm việc với nhau dễ dàng hơn. Thuận tiện trong việc phát triển hệ thống lớn
* Có kiểu dữ liệu rõ ràng , chặt chẽ .
* Lỗi có thể được phát hiện ngay lúc biên dịch.
* Hỗ trợ hướng đối tượng .

1. Làm thế nào để lấy được dữ liệu từ component cha

* Sử dụng decorator @Input ở thư viện angular/core để khai báo cho thuộc tính nhận dữ liệu từ component cha ở component con

1. Làm thế nào để lấy được dữ liệu từ component con

* Tạo đối tượng EventEmitter . Gắn @Output cho đối tượng này. gọi phương thức emit() của event emitter vào thời điểm cần tung event, sử dụng event binding ở component cha để hứng và bắt dữ liệu.

1. Sự khác biệt giữa RouterModule.forChild và RouterModule.forRoot là gì?

* ForRoot : load tất cả các component ngay lúc start angular
* ForChild : giúp triển khai cơ chế lazy loading -> khi nào gọi thi component mới load

1. Tại sao nên sử dụng Angular (điểm mạnh)?

* Vì angular là framework nên nó có cấu trúc rõ ràng dựa vào các class , dependency và mô hình MVVM. => giúp các thành viên thống nhất code với nhau, build được trang web nhanh hơn.
* Là ứng dụng SPA . Nâng cao tính trải nghiệm của người dùng.
* Mạnh về xử lí bất dồng bộ.
* Sử dụng Typescript -> bắt được lỗi ngay lúc compile
* Có JIT

1. Nhược điểm của Angular là gì?

* Vì được viết = typescript cũng chính là superset của JS nên độ bảo mật , an toàn kém . Nếu bị tắt script thì chỉ thấy được những thành phần cơ bản của HTML

1. Angular Cli là gì? Làm sao để sử dụng nó ?

* là công cụ được phát triển để chạy trên nền cửa sổ ứng dụng nhằm giúp đỡ việc tạo dự án , quản lí tập tin trong dự án và thực hiện các công việc khác như kiểm thử , deploy dự án nhanh và hiệu quả

1. Directive trong angular là gì ?

* Là các thành phần sử dụng để thay đổi cấu trúc template trong component
* Bao gồm :
* Component directive chính là component thường dùng.
* Structural \*ngfor \*ngIf \*ngSwitch
* Attribute directive : [ngStyle] [ngClass]

1. Trình bày được khái niệm template trong angular

* Mỗi 1 component sẽ gắn liền với 1 template , component có nhiệm vụ xử lí logic của template và tempalte HTML sẽ có nhiệm vụ mô tả giao diện

1. Giải thích kiến trúc của Angular

* Component : là các thành phần xử lí giao diện và hiển thị giao diện lên view. Bao gồm :
* Template : lay out / giao diện hiển thị của component.
* Class ( properties + method ) : xử lí nghiệp vụ logic của component
* Metadata : chính là phần định nghĩa component về tên gọi , đường dẫn đến HTML , CSS nằm trong file . ts.
* Cơ chế databinding giữa View viewmodel.
* Service là các thành phần hỗ trợ cho view
* Directive là các thành phẩn hỗ trợ thay đổi cấu trúc của template trong component . Có 3 loại : component directive , structural directive , attribute directive.
* Module bao gồm các thành phần component + service + routing

1. Hãy mô tả khái niệm Databinding One Way & two way phân biệt sự khác nhau?

* One way : bind 1 chiều .từ viewmodel -> view ( interpolation,property binding) . từ view -> viewmodel ( event binding)
* Two way: binding 2 chiều

1. ngModule là gì?

* Module là cách chia nhỏ các thành phần trong ứng dụng . @ngModule là decorator để khai báo đó là module
* Bao gồm :
  + Import : là mảng các module dùng trong module này
  + Declaration : Khai báo các component,directive..dùng trong module này
  + Bootstrap: định nghĩa module gốc khởi động đầu tiên
  + Provider: khai báo service cần DI vào các component hay toàn bộ module

1. Component là gì ? các thành phần của component

* Component là thành phần giao diện trên view .
* Bao gồm :
* Template : là giao diện hiển thị lên view của component
* Class : gồm method + properties . xử lí logic của component
* Metadata : chính là thành phần định nghĩa cho component về tên gọi , đường dẫn đến HTML , CSS

1. Có thể chuyển mã TypeScript thành mã Javascript không, bằng cách nào?

* Có thể . dùng cơ chế transpile

1. Mô tả vòng đời component?

* ngOnchange sẽ được gọi khi component phát hiện có giá trị được binding vào component = input property ( từ cha -> con).
* ngOnInit sẽ được gọi khi component được tạo ra lần đầu tiên . được gọi sau onchange
* ngOnDocheck được gọi khi phát hiện ra có sự thay đổi dữ liệu trong component
* ngOndestroy : được gọi khi component bị phá hủy.

1. Pipe là gì?

* Là thành phần được sử dụng để format lại kiểu hiển thị trên view

1. Phân biệt sự khác nhau giữa Promise và Observable

|  |  |
| --- | --- |
| Observable | Promise |
| * Trả về nhiều kết quả 1 lúc * Hủy được request * Cung cấp retry/retryWhen có thể connect lại API khi gặp sự cố * Có khả năng xử lí kết quả trước khi subcribe | * Trả về 1 kết quả duy nhất * Không hủy được. * Không thể retry * Chỉ xử lí được kết quả sau khi trả về |

1. Biên dịch AOT là gì? Những lợi thế của AOT ?

* AOT viết tắt của Ahead of Time : biên dịch trong lúc đang build chương trình
* Lợi thế :
  + Khởi động ứng dụng lần đầu nhanh .
  + Lập trình viên chủ động trong việc biên dịch .
  + Thích hợp cho quá trình sử dụng
  + Khi có lỗi sẽ dừng chương trình và show lỗi

1. Trình bày khái niệm Front-end & backend

* Front-end là phần giao diện của trang web tương tác với người dùng .Được thiết kế bở HTML CSS JS và các front-end framework
* Back-end là phần bao gồm máy chủ , ứng dụng , cơ sở dữ liệu v.v.. nơi sẽ xử lí nghiệp vụ với dữ liệu và cơ sở dữ liệu.

1. Làm sao để cài đặt TypeScript

* Tạo file tsconfig.json
* Tạo package.json
* Chạy câu lệnh npm i
* vào src khởi tại file typescript

1. Để khởi tạo một dự án angular sử dụng câu lệnh nào

* ng new project-name

1. Để chạy một dự án angular sử dụng câu lệnh nào

* Ng s - o / ng s

1. Kể tên các kiểu dữ liệu trong Typescript

* Tất cả các kiểu dữ liệu của js , interface , enum , any , tuple

1. Kể tên các vòng lặp được typescript hỗ trợ

* Vòng lặp for : for , for in , for of
* Vòng lặp while
* Vòng lặp do-while

1. ngFor là gì? Cách sử dụng

* Là Structural directive hỗ trợ vòng lặp ở template của angular

1. ngIf là gì? Cách sử dụng

* Là structural directive hỗ trợ điều kiện ở template của angular

1. Các cách tạo form trong Angular

* Template driven form : tạo form dựa vào template = cách thêm các directive vào . sau đó angular sẽ tự động tạo và quản lí. Khai bảo ở thẻ form 1 template variable =’ngForm’. trong thẻ input khai báo ngModel đi kèm attribute name.
* Model driven form ( reactive form) : tạo form = cách tạo các object model ở component rồi tạo form từ chúng ( form group ,form array , form control)

1. router trong angular dùng để làm gì?

* Là thành phần giúp chúng ta điều hướng đến các component tương ứng
* Các template của component sẽ được hiển thị qua router outlet

1. Cách để lấy tham số trên URL trong angular

* Tiêm activated route ở constructor
* Sử dụng snapshot.params.<parameter\_name> hoặc paramMap.subcribe

1. làm sao để kiểm tra người dùng có quyền truy cập vào 1 route ?

* Sử dụng router-guard để phân quyền ở angular

1. lazy-loading là gì?

* Là cơ chế mà chỉ load khi component khi cần thiết .