Item 1: Reutilização de componentes

a) A respeito de componente, responda:

I. O que é um componente em aplicações Angular?

É qualquer coisa visível para o usuário e pode ser utilizado várias vezes em um aplicativo.

II. Do que um componente é composto?

De um seletor, um teplante (HTML) e um style (CSS).

III. Qual é o comando utilizado para criar um componente utilizando Angular CLI?

ng g c <nome do componente>.

IV. Qual a importância da reutilização de componentes?

Ao criar componentes reutilizáveis você consegue otimizar o trabalho,fica mais fácil de fazer ajustes, corta custos desnecessários, facilita a colaboração.

V. Qual a funcionalidade do selector no exemplo abaixo?

Exemplo: @Component({
selector: 'app-component-overview',
})

É como identificamos o nosso componente. Para todo o componente existe um elemento único associado que permite que ele seja adicionado em um documento HTML, no caso <app-component-overview>

VI. Explique a funcionalidade de templateUrl e template nos exemplos abaixo e quando devemos utilizar cada uma delas.

Exemplo 1: @Component({
selector: 'app-component',
templateUrl: './component.component.html'
})

É o nome do documento HTML que será a parte visual do componente, utilizado quando temos um código mais elaborado e maior. Nesse caso esta sendo indicado o local do HTML.

Exemplo 2: @Component({
selector: 'app-component-overview',
template: '<h1>Hello World!</h1>'
})

Usado para descrever também a parte visual do componente, utilizado quando o código contém poucas linhas, seguindo as boas práticas do angular.

VII. Explique a funcionalidade de styleUrls e styles nos exemplos abaixo e quando devemos utilizar cada um deles.

Exemplo 1: @Component({
selector: 'app-component',
templateUrl: './component.component.html',
styleUrls: ['./component.component.css']
})

Mesmo conceito do templateUrl, porém utilizando o estilo que será aplicado ao template.

Exemplo 2: @Component({
selector: 'app-component-overview',
template: '<h1>Hello World!</h1>',
styles: ['h1 { font-weight: normal; }']
})

Mesmo conceito do template. Usado no próprio TS quando temos poucas linhas de estilo.

Hands-on

Completado.

Reutilizando códigos através de herança

Completado.

Reutilizando componentes através do seletor

Completado.

Exercício prático

Utilizando o que foi aprendido até agora, através da reutilização de códigos e componentes:

Completado.

Item 2: Rotas

Sobre o funcionamento das rotas do Angular, responda:

I. Qual é o comando no Angular CLI para criar uma nova aplicação com rotas?

ng new [nome-do-projeto] -- routing

II. Qual é o comando no Angular CLI para criar um novo módulo com rotas?

ng generate module [nome-do-projeto] -- routing.

III. Para que serve o método navigate( ) da classe Router?

Serve para navegar entre os componentes sem a necessidade de recarregar a aplicação toda vez que muda de página.

IV. Para que serve o método isactive( ) da classe Router?

Serve para determinar a rota ativa, a página atual em que o usuário esta navegando.

V. Para que serve o arquivo src/app/app-routing.module.ts?

É o arquivo de módulo das rotas, ele é importado para o root module e contém todos os arquivos necessários para criarmos rotas.

VI. Para que serve a tag <router-outlet></router-outlet>?

RouterOutlet é uma das diretivas de Router. É ele que faz o direcionamento da rota para seu component. correspondente.

VII. Dê um exemplo de um botão chamando uma rota através de um método em angular.

```
<button (click)="cubo()">
  Quanto é {{ numeroComponent }} ao cubo?
</button>
```

VIII. Dê um exemplo de um botão chamando uma rota sem chamar um método em angular.

```
<button [routerLink]="['/quadrado']" [state]="{ valorRota: numeroComponent }">
  Quanto é {{ numeroComponent }} ao quadrado?
</button>
```

IX. O que é Lazy Load? Dê um exemplo da definição de uma rota em angular utilizando Lazy Load.

O lazy load é uma forma de aumentar a performance da sua aplicação, de modo que um módulo e seus componentes só serão carregados se eles forem chamados.

Exemplo:

Uma página de ‘cadastro’ contendo os botões ‘logar’ e ‘esqueci minha senha’

a rota só será chamada caso o usuário clique em algum botão.

X. O que é Eager Load? Dê um exemplo da definição de uma rota em angular utilizando Eager Load.

Eager Load é o carregamento padrão do angular, todas as rotas são carregadas e ficam prontas para serem chamadas pelo usuário.

Exemplo:

Uma página de ‘cadastro’ contendo os botões ‘logar’ e ‘esqueci minha senha’

As rotas para cada botão já foram carregadas antes no início da aplicação.

Hands-on

Completado.

Exercício Prático

Criar um novo componente que mostre o resultado de X^3, onde o X deve ser passado através de rotas.

Completado

Item 3: Pipes

I. Qual é a utilidade dos pipes em aplicações Angular?

Transformar dados antes de serem visualizados no template.

II. Cite os 6 pipes que fazem parte do pacote inicial do Angular.

UpperCasePipe**,** DecimalPipe, DatePipe, CurrencyPipe, LowerCasePipe, PercentPipe

III. Descreva a funcionalidade de cada um dos pipes citados acima e dê um exemplo da sua utilização, juntamente com o resultado em tela.

UpperCasePipe - Transforma todo o texto em letras maiúsculas.

Exemplo: {{ 'Título da página' | uppercase }}

TÍTULO DA PÁGINA

DecimalPipe - Transforma um número e suas casas decimais de acordo com o padrão da região.

Exemplo: {{ 12345678 | number : ‘ ‘ : ‘pt-BR’ }}

1.234,568

DatePipe - Transforma a data de acordo com a região.

Exemplo: Data do Evento: {{ evento.data | date: ‘shortDate’ }}

Data do Evento: 06/02/2022

CurrencyPipe - Transforma um número em moeda local.

Exemplo: Preço: {{ evento.preço | currency: ‘BRL’ }}

Preço: R$ 15.00

LowerCasePipe - Transforma todo o texto em letras minúsculas.
Exemplo: {{ ‘Subtítulo da página’ | lowercase }}
subtítulo da página

PercentPipe - Transforma um número em porcentagem de acordo com o padrão da região.
Exemplo: {{ 0.1234 | percent: ‘1.2-2’ }}
12,34%

IV. O que são custom pipes?

São pipes personalizados seguindo o padrão de formatação que o desenvolvedor precisar para transformar valores.

Hands-on

Completado

Exercício Prático

Completado