**Angular Avançado (CRUD)**

**O que o método filter do RxJS faz?**

**R:** Nos retorna somente os registros que obedecem a condição previamente estabelecida.

**O Angular Material já vem instalado por padrão nos projetos Angular?**

**R:** Não, nós precisamos instalar ele no nosso projeto.

**Qual é a função do pipe no RxJS?**

**R:** Ele faz com que seja possível chamar métodos do RxJS antes do retorno dos dados.

**O que é o Angular Material?**

**R:** É uma implementação do Material Design da Google.

**É possível encontrar exemplos de como utilizar o Angular Material no site da biblioteca?**

**R:** Sim, no site do angular material possui vários exemplos de utilização.

**Eu preciso chamar o método subscribe para que um Observable execute a sua ação?**

**R:** Sim, o subscribe faz com que você se inscreva no Observable, e a partir desse momento o Observable vai executar sua ação.

**Ao utilizarmos os componentes do Angular Material não podemos utilizar nenhum outro componente?**

**R:** Podemos utilizar qualquer outro componente, o Angular Material apenas nos entrega alguns componentes pronto para agilizar o nosso desenvolvimento.

**O RxJS já vem instalado nos projetos Angular?**

**R:** Sim, não precisamos fazer nenhuma configuração para ele funcionar.

**O que é o RxJS?**

**R:** O RxJS é uma biblioteca para programação reativa usando Observables.

**O que o método map do RxJS faz?**

**R:** Permite que nós manipulamos os valores do retorno do Observable.

--------------------------------------------------------------------------------------------

**Estrutura de Projeto exemplo**

**O que acontece quando não declaramos um componente dentro de um module?**

**R:** O nosso projeto fica com erro e não vai buildar, pois sem um módulo o Angular não sabe as dependências do componente.

**Qual a função do JSON-Server?**

**R:** Criar chamadas REST para um back-end mock.

**Ao baixar o projeto já vem todas as dependências instaladas?**

**R:** Não, o projeto vem com o arquivo package.json, onde nós executamos o comando npm install e aí sim vai ser instalado às dependências do projeto.

**O nosso projeto inicial possui alguma coisa pronta?**

**R:** Sim, ele já tem uma estrutura simples para que nós não precisemos ficar criando código html básico.

**Qual a função do nosso arquivo app.routing.module.ts?**

**R:** Ele é o responsável por todas as rotas do nosso sistema.

**Por que as dependências não vêm baixadas no projeto?**

**R:** Pelo fato de que são muitos arquivos e isso deixa nosso projeto muito pesado para ser baixado.

**Qual o comando utilizado para iniciar a nossa aplicação Angular?**

**R:** npm start

**Qual a função do arquivo app.module.ts?**

**R:** Ele é o arquivo principal da nossa aplicação, todo o carregamento do projeto inicia por ele.

**Por que instalamos o JSON-Server com o parâmetro -g?**

**R:** O parâmetro -g faz com que o pacote seja instalado globalmente.

**O nosso projeto possui algum padrão de nomenclatura para os componentes seletores?**

**R:** Sim, todos os componentes devem iniciar com a sigla ‘dio’ que significa Digital Innovation One.

--------------------------------------------------------------------------------------------

**Criando Formulários Reativos**

**É possível passarmos propriedades para os componentes que nós criamos?**

**R:** Sim, só precisamos colocar nos nossos componentes a notação @Input() nome-da-propriedades e quando chamarmos nosso componente passamos o valor desejado ali.

**Para que serve o FormBuilder?**

**R:** É com ele que criamos os nossos formulários reativos, passamos por parâmetro os campos e ele já retorna um FormGroup pronto para uso.

**No nosso formulário HTML qual a propriedade que faz referência ao campo input com o formulário no arquivo TS?**

**R:** formControlName

**Como saber que um campo está com erro?**

**R:** Cada campo possui uma propriedade chamada erros e nela tem um array com todos os erros do campo que estão acontecendo.

**Por questão de organização, qual tipo de método adicionamos primeiro?**

**R:** Primeiramente adicionamos todos os métodos públicos depois os privados.

**Nós conseguimos colocar validações nos formulários?**

**R:** Sim, só precisamos passar as validações desejadas por parâmetro na criação do formulário.

**Qual a função do Elvis Operator?**

**R:** Elvis Operator ou operador de navegação segura serve para garantir que se a variável no HTML estiver null ou undefined o Angular não tente acessar o valor dela.

**Por que nós componentizamos as coisas no Angular?**

**R:** Para re-utilizarmos os componentes, assim não geramos códigos duplicados.

**O que acontece se não especificarmos a propriedade formGroup em um formulário HTML?**

**R:** O Angular não vai saber qual formulário está sendo utilizado nos campos e vai dar erro.

**É possível passarmos um valor inicial para os campos do nosso formulário?**

**R:** Sim, o primeiro parâmetro de cada campo na criação do formulário é o responsável pelo valor inicial de cada input.

**Criando Serviços e Tratamento de Retorno Modal**

**Nós utilizamos modal para apresentar mensagens aos clientes. De qual biblioteca nós pegamos esse modal?**

**R:** Angular Material

**O que é o HttpCliente?**

**R:** Serviço responsável por criar nossas chamadas REST.

**Por que nós criamos interfaces?**

**R:** Elas garantem que os objetos (contratos) sigam um padrão estabelecido, garantindo que todas os desenvolvedores sigam o mesmo padrão.

**Qual das opções abaixo seria uma forma válida de pegar os valores de um formulário do tipo FormGroup?**

**R:** getRawValue()

**Qual o método HTTP que nós utilizamos para salvar um registro?**

**R:** POST

**Alguns componentes precisam ser instanciados já quando a nossa aplicação é carregada. Para esses componentes nós declaramos em um local específico do modulo. Que lugar é esse?**

**R:** entryComponents

**Como fazemos para exibir o modal somente depois do cadastro ter sido realizado?**

**R:** Colocamos o método de abertura do modal dentro do subscribe.

**Qual o comando do angular CLI para criar um serviço?**

**R:** ‘ng g s nome-serviço ou ‘ng generate service nome-serviço

**Como podemos especificar o tipo de uma variável?**

**R:** Utilizando dois pontos ‘:’ ou ‘as’.

**Por questão de organização, qual é a melhor forma de ordenarmos os imports dentro do Angular?**

**R:** Primeiro os imports do Angular, depois de terceiros e por último os nossos.

--------------------------------------------------------------------------------------------

**Listagem e Melhoria de Perfomance**

**Qual biblioteca usamos para criar um scroll infinito?**

**R:** NGX-Infinite-scroll

**Qual método a biblioteca de scroll nos fornece para sermos avisados que o usuário fez scroll na página?**

**R:** scrolled

**O que podemos fazer para que o HTML entenda as quebras de linha do campo descrição?**

**R:** Podemos criar uma classe CSS com white-spece: pre-line

**Como fazemos para repetir um trecho de código no HTML para exibir nossa lista de filmes?**

**R:** Utilizamos o \*ngFor.

**Como concatenamos dois array?**

**R:** Utilizando o método push, porém precisamos também adicionar o spread operator.

**Como passamos parâmetros para o HttpParams?**

**R:** Através do método set.

**Como podemos usar um Else no HTML?**

**R:** Colocando o comando else dentro do \*NgIf e usando a tag ng-template.

**O que é o HttpParams?**

**R:** É uma classe responsável por passar parâmetros em consultas REST.

**É possível reduzir o número de requisições que fazemos ao back-end no filtro do campo texto que inserimos na listagem de filmes?**

**R:** Sim, podemos utilizar debounceTime do RxJS.

**Qual método HTTP nós utilizamos para buscar os filmes?**

**R:** GET

--------------------------------------------------------------------------------------------

**Finalizando Projeto**

**Qual o método HTTP utilizado para editar os filmes?**

**R:** PUT

**Como podemos transformar uma data para o formato brasileiro?**

**R:** O Angular possui um pipe pronto para isso, basta passar o pipe date e o formato desejado.

**Qual classe utilizamos para obter os parâmetros da rota?**

**R:** ActivatedRoute

**É possível passarmos parâmetros através das rotas?**

**R:** Sim, para isso precisamos alterar a rota colocando cada parâmetro iniciando com dois pontos ‘:’.Exemplo: ‘:id-do-filme’ para que a rota entenda que aquilo é um parâmetro.

**O que acontece se tentarmos acessar uma URL inexistente no nosso projeto?**

**R:** Seremos redirecionados para a listagem dos filmes, pois especificamos isso no arquivo de rotas.

**Qual o método HTTP utilizado para excluir os filmes?**

**R:** DELETE

**Como sabemos que estamos editando um filme e não cadastrando um novo?**

**R:** Quando estamos editando nós recebemos o ID do filme para edição e quando estamos cadastrando esse ID é null.

**O que vai acontecer se não passarmos o ID do filme para edição?**

**R:** Não vai ser possível editar e vai ocorrer um erro na requisição pois estamos tentando fazer uma requisição PUT sem passar o parâmetro de identificação.

**É possível duas rotas usarem o mesmo componente?**

**R:** Não existe limite de uso para um componente, basta apenas colocar na rota qual é o componente referente a rota.

**Se adicionarmos uma classe CSS no arquivo CSS de um componente, em quais lugares conseguimos utilizar essa classe?**

**R:** Somente no HTML do próprio componente.