



Angular 9

Aula #1

Vídeo Aula e Exercícios

Esta apostila é referente a vídeo aula #1 e a sequência de exercícios #1 do curso Angular 9. Não deixe de assistir a vídeo aula e fazer os exercícios propostos para esta aula em nossa Plataforma Online.

Sobre os materiais da Grande Porte

Todos os materiais desenvolvidos pela Grande Porte objetivam ensinar de maneira eficiente, explorando potenciais e sanando dificuldades. As atualizações são feitas constantemente para manter o conteúdo atualizado com as demandas de mercado e tendências para o futuro. Esperamos que você aproveite este material. Comentários, críticas e sugestões serão muito bem-vindos.

ATENÇÃO

Os materiais desenvolvidos pela Grande Porte são distribuídos através da Plataforma Online apenas para assinantes. Todos os direitos são reservados à Grande Porte. A distribuição, cópia, revenda e utilização para ministrar treinamentos, sem autorização, são absolutamente proibidas. Se você deseja obter autorização para usar nossos materiais comercialmente, por favor, entre em contato.

Introdução ao Angular 9

O Angular é um framework (plataforma de desenvolvimento) open source (código aberto) para o desenvolvimento de aplicações web, baseada em HTML, CSS e TypeScript. Foi desenvolvida por Mohammad Shoman e seu time na Google em conjunto com uma comunidade de pessoas e empresas que constantemente ajudam em sua melhoria, para atender aos anseios dos desenvolvedores.

O Angular é também chamado de Angular 2+ ou angular 2 (ou superior), para não ser confundido com o AngularJS.

Como veremos a seguir, a cada 6 meses aproximadamente tivemos uma nova versão do Angular e, segundo a Google, isso vai continuar acontecendo.

Este curso é sobre o Angular 9.

Diferenças entre AngularJS e Angular 9

A primeira versão do Angular (Angular 1.0) foi originalmente desenvolvida em 2009 por Miško Hevery e Adam Abrons como um software por trás de um serviço de armazenamento JSON online, que teria preço estimado por megabyte, para aplicações pré-construídas (easy-to-make) para as empresas. Este empreendimento foi disponibilizado em "GetAngular.com", e teve alguns inscritos, antes de os dois decidirem abandonar a ideia comercial e distribuir a primeira versão do Angular como um framework open-source.

Adam Abrons deixou o projeto, mas Miško Hevery, que trabalha no Google continuou seu desenvolvimento a partir de 2010, mantendo o framework em conjunto com alguns colegas do Google: Igor Minár e Vojta Jína.

Quando foi reescrita, a nova versão do Angular (Angular 2.0) foi chamada de Angular 2. Mas ela era tão diferente da versão 1.0 que causou confusão entre os desenvolvedores, pois como pode a versão 2 de uma linguagem ser incompatível com a versão 1?

Para resolver este problema, a equipe resolveu rebatizar a versão 1.0 para AngularJS e as novas versões passariam apenas a se chamar de Angular, desde o Angular 2 até o atual Angular 9.

Então quando alguém disser que programa em Angular, você já sabe que não é o AngularJS.

A evolução do Angular foi a seguinte, resumidamente:

- **Setembro de 2014 - Angular 2**

As mudanças drásticas na versão 2.0 criou uma considerável controvérsia entre os desenvolvedores, levando a alterar o nome da versão 1.0 para Angular JS.

- **Dezembro de 2014 - Angular 4**

O Angular 4 foi lançado, ignorando o Angular 3 para evitar uma confusão devido ao desalinhamento da versão do pacote do roteador que já foi distribuído como v3.3.0.

- **Novembro de 2017 - Angular 5**

Os principais aprimoramentos no Angular 5 incluem suporte para web apps progressivos, uma compilação otimizadora e melhorias relacionadas ao Material Design.

- **Maior de 2018 - Angular 6**

Esta versão teve um foco menor na base do framework, e maior na cadeia de ferramentas e em como tornar mais fácil e rápida a migração com Angular nas atualizações futuras.

- **Outubro de 2018 - Angular 7**

Atualizações relacionadas ao desempenho de aplicativos, Angular Material, Rolagem Virtual etc.

- **Maior de 2019 - Angular 8**

Conta com algumas melhorias no funcionamento dos formulários reativos, melhoria na sintaxe para declaração de rotas lazy-loading e a primeira versão do compilador Ivy.

- **Fevereiro de 2020 - Angular 9**

Agora com uma versão sofisticada do compilador Ivy como default, melhorias no Angular forms, mudanças na injeção de dependências do núcleo entre muitas outras.

Requisitos para fazer este curso

Como o objetivo do Angular é o desenvolvimento de aplicações web, há alguns pré-requisitos que precisam ser observados para que o aluno possa ter total desenvolvimento e aprendizado neste curso.

Esses pré-requisitos não são impeditivos, pois durante o treinamento explicaremos cada um dos comandos e tags utilizadas, mas é importante que o aluno procure se aprofundar nesses pré-requisitos, que são:

HTML (Hyper Text Markup Language ou Linguagem de Marcação de Hiper Texto) – é a linguagem utilizada para dar estrutura às páginas web. É composta por tags de formatação de texto e é muito simples de ser aprendida.

CSS (Cascading Style Sheets ou Folhas de Estilo em Cascata) – provê um mecanismo de adicionar estilo, tais como cores, fontes, espaçamento etc. em uma página web.

Javascript – é uma linguagem de programação (estruturada e orientada a objetos) utilizada para dar dinamismo às páginas web.

Typescript – é um superconjunto de Javascript. É uma linguagem de programação que acrescenta tipo e outros recursos ao Javascript. Sua compilação gera Javascript.

Durante este curso de Angular 9, faremos uso dessas linguagens e estaremos explicando pontualmente sua utilização.

SPA (Single Page Application)

Uma SPA (Single Page Application ou Aplicativo de Página Única) é um aplicativo ou site da Web que interage com o navegador (Browser), reescrevendo dinamicamente a página atual com novos dados do servidor da Web, em vez do método padrão do navegador carregar páginas inteiras. O objetivo são transições mais rápidas que fazem o site parecer mais um aplicativo nativo.

Em um SPA, todo o código HTML, JavaScript e CSS necessário é recuperado pelo navegador com um único carregamento de página ou os recursos apropriados são carregados dinamicamente e adicionados à página conforme necessário, geralmente em resposta às ações do usuário.

A página não é recarregada em nenhum momento do processo, nem controla a transferência para outra página. Uma aplicação web SPA se assemelha muito a uma aplicação desktop ou mesmo mobile.

Uma típica aplicação web SPA possui, conforme figura 1.a, uma barra lateral e outra superior, que podem por exemplo, serem utilizadas como menus. O bloco verde, é a área central onde ocorre a troca de conteúdo de forma dinâmica. O usuário não sente a troca de página, pois a página em si não é recarregada, apenas o conteúdo da área central é atualizado.

Já na figura 1.b, temos uma aplicação web tradicional, onde a partir da primeira página, quando há a necessidade de mostrar novas informações, a página é recarrega, fazendo com que o usuário perceba que a página inteira está sofrendo atualização.

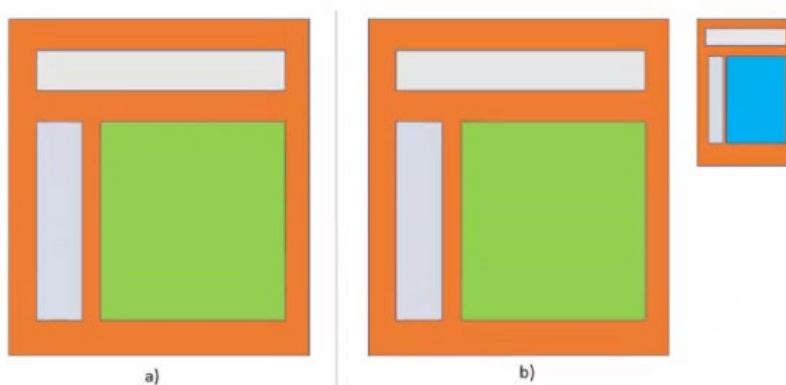


Figura 1 - A primeira imagem (a) é uma aplicação SPA, na qual passa-se a impressão de que a aplicação roda em apenas uma tela. A segunda imagem (b) é uma aplicação na qual possui várias páginas.

A experiência do usuário em aplicações SPA é melhor em relação às não-SPA, daí a sua importância atualmente para o desenvolvimento de novas aplicações web.

Baseado em Componentes

O Angular é baseado em componentes e módulos. Para o angular, tudo é ou um componente ou um módulo.

Os componentes são basicamente classes definidas para serem visíveis em qualquer elemento na tela ou nos navegadores. Um componente interage com seu arquivo.html, enviando e recebendo dados.

Podemos dizer, conforme figura 2, que o menu superior é um componente, o menu lateral é um componente e na área central (verde) temos diversos componentes, pois um componente pode herdar de outro componente ou mesmo chamar outro componente.

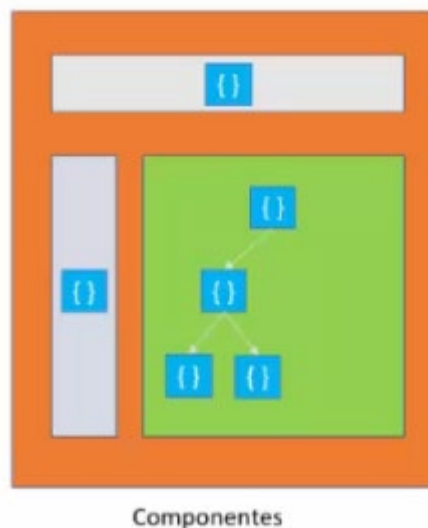


Figura 2 – Uma aplicação e seus componentes.

Um componente está ligado à uma parte da visualização, daquilo que o usuário vê (HTML) e um conjunto de componentes formam uma aplicação.

Os módulos são contêineres para um bloco de código coeso dedicado a um domínio de aplicativo, para organizar componentes, diretivas, serviços, pipes e outros elementos relacionados à aplicação.

Na figura 3 temos uma área verde e outra amarela, representando dois módulos da aplicação. Cada componente deve estar dentro de um módulo e um componente pode ser compartilhado por mais de um módulo, sendo essa situação muito comum.

Você pode, por exemplo usar um componente do módulo amarelo dentro do módulo verde, uma situação de caracteriza a reutilização de componentes no Angular 9.

Neste curso você aprenderá, nas próximas aulas, como organizar sua aplicação em módulos.

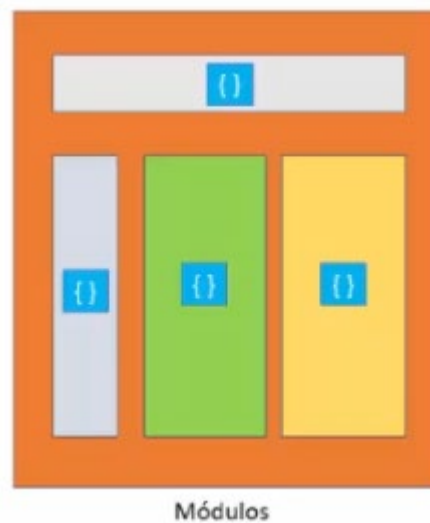


Figura 3 – Uma aplicação e seus módulos.

Vídeo Aula e Exercícios

Esta apostila é referente a vídeo aula #1 e a sequência de exercícios #1 do curso Angular 9. Não deixe de assistir a vídeo aula e fazer os exercícios propostos para esta aula em nossa Plataforma Online.

Sobre o autor

Fabrizio Borelli



Doutorando em Engenharia da Informação pela UFABC

Mestre em Ciência da Computação pela UFABC

Bacharel em Ciência da Computação pela UFABC

Professor Universitário

Empreendedor em TI