

Laboratório Prático: Trabalhando com Iteração de Array para Lista de Menu de Restaurante



Tempo estimado necessário: 20 minutos

O que você vai aprender

Neste laboratório, você explorará como o JavaScript gera dinamicamente seções de menu de restaurante para café da manhã, prato principal e sobremesa usando os métodos de array do JavaScript. Você aprenderá a usar map para preencher o menu de café da manhã, forEach para exibir os itens do menu do prato principal e um loop for tradicional para renderizar os itens de sobremesa. Além disso, você entenderá como o conteúdo HTML pode ser atualizado dinamicamente, demonstrando o poder do JavaScript em modificar elementos de página da web com base em arrays predefinidos.

Objetivos de aprendizado

Após completar este laboratório, você será capaz de:

- **Compreensão da manipulação do DOM:** Ganhar proficiência em acessar elementos HTML e atualizar dinamicamente seu conteúdo usando JavaScript. Focar no método getElementById para alterar precisamente elementos específicos dentro de uma página da web.
- **Métodos de iteração de array:** Diferenciar entre vários métodos de iteração de array, como map, forEach e loops for. Observar suas funcionalidades distintas ao iterar por arrays e gerar conteúdo HTML estruturado.
- **Criação de conteúdo web dinâmico:** Aprender o processo de gerar e renderizar dinamicamente conteúdo em uma página da web usando JavaScript para criar menus estruturados com base em arrays predefinidos, enfatizando a natureza dinâmica do desenvolvimento web.
- **Organização das seções do menu:** Compreender o conceito de organizar o conteúdo da página da web em seções (por exemplo, café da manhã, prato principal, sobremesa) populando elementos HTML com dados de arrays JavaScript correspondentes, facilitando layouts de página da web estruturados e legíveis.
- **Aplicação de métodos JavaScript:** Aplicar métodos JavaScript (map, forEach, getElementById, innerHTML) para manipular arquivos HTML de forma eficaz, demonstrando o uso prático desses métodos na geração e atualização de conteúdo da página da web dinamicamente com base em dados predefinidos.

Pré-requisitos

- Conhecimento básico de HTML.
- Navegador web com console (Chrome DevTools, Firefox Console e mais).

Passo 1: Configurando o ambiente

1. Primeiro, você precisa clonar seu repositório principal no **Ambiente Skills Network** que você criou no primeiro laboratório e onde você enviou todos os arquivos e pastas dos seus laboratórios anteriores. Siga os passos dados:
 - Para isso, clique no terminal no painel da janela à direita e selecione **Novo Terminal**.
 - Execute o comando git clone escrevendo o comando dado no terminal.

```
git clone <github-repository-url>
```

Nota: Coloque o link do seu próprio repositório GitHub em vez de <github-repository-url>.

- O passo acima irá clonar a pasta do seu repositório GitHub dentro da pasta do projeto no explorador. Você também verá várias pastas dentro da pasta clonada.
- Agora você precisa navegar para dentro da pasta clonada. Para isso, escreva o comando dado no terminal:

```
cd <repository-folder-name>
```

Nota: Escreva o nome da sua pasta clonada em vez de <repository-folder-name>. Execute `git clone` se você tiver feito logout do **Ambiente Skills Network** e não conseguir ver nenhum arquivo ou pasta após fazer login novamente.

2. Agora, selecione a pasta **cloned Folder Name**, clique com o botão direito nela e clique em **Nova Pasta**. Digite o nome da pasta como **restaurantMenu**. Isso criará a pasta para você. Em seguida, selecione a pasta **restaurantMenu**, clique com o botão direito e selecione **Novo Arquivo**. Digite o nome do arquivo como **restaurant_menu.html** e clique em **OK**. Isso criará seu arquivo HTML.
3. Agora selecione a pasta **restaurantMenu** novamente, clique com o botão direito e selecione **Novo Arquivo**. Digite o nome do arquivo como **restaurant_menu.js** e clique em **OK**. Isso criará seu arquivo JavaScript.
4. Crie a estrutura básica do arquivo HTML adicionando o conteúdo fornecido abaixo.

```
<!DOCTYPE html>
<html>
<head>
    <title>Cardápio do Restaurante</title>
</head>
<body>
    <h1>Iteração do Cardápio do Restaurante</h1>
        <h2>Cardápio de Café da Manhã</h2>
        <div id="breakfastTotalItems"></div>
        <div id="breakfastMenuItems"></div>
        <h2>Cardápio de Prato Principal</h2>
        <div id="maincourseTotalItems"></div>
        <div id="maincourseMenuItems"></div>
        <h2>Cardápio de Sobremesas</h2>
        <div id="dessertTotalItems"></div>
        <div id="dessertMenuItems"></div>
        <script src=".//restaurant_menu.js"></script>
</body>
</html>
```

- Este código HTML fornece a base estrutural para exibir um cardápio de restaurante em três seções: café da manhã, prato principal e sobremesa. Ele contém elementos como `<h1>` para o título geral, tags `<h2>` para cada seção do cardápio e elementos `<div>` com IDs únicos (`breakfastTotalItems`, `breakfastMenuItems`, `maincourseTotalItems`, `maincourseMenuItems`, `dessertTotalItems`, `dessertMenuItems`) destinados a exibir dinamicamente o conteúdo do cardápio.
- No entanto, até agora, não há conteúdo real do cardápio sendo populado ou exibido dinamicamente. Código JavaScript é necessário para gerar e inserir itens do cardápio nos respectivos elementos `<div>` identificados por seus IDs únicos.

Nota: Quando você tiver colado o código, salve seu arquivo.

Passo 2: Definindo variáveis e método de array map

1. No **restaurant_menu.js**, inicialize três arrays nomeados como **breakfastMenu**, **mainCourseMenu** e **dessertMenu**. Inicialize esses arrays com os itens do menu conforme dado no código abaixo. Inclua este código no arquivo **restaurant_menu.js**.

```
const breakfastMenu = ['Pancakes', 'Eggs Benedict', 'Oatmeal', 'Frittata'];
const mainCourseMenu = ['Steak', 'Pasta', 'Burger', 'Salmon'];
const dessertMenu = ['Cake', 'Ice Cream', 'Pudding', 'Fruit Salad'];
```

2. Agora, usaremos um loop `for` e dois métodos de array: o método **map** e o método **forEach** para percorrer esses arrays e exibir os itens do menu na página HTML.
3. Primeiramente, vamos iterar pelo array **breakfastMenu** usando o método **map**. Inclua o seguinte código no arquivo **restaurant_menu.js**.

```
const breakfastMenuItemsHTML = breakfastMenu.map((item, index) => `<p>Item ${index + 1}: ${item}</p>`).join('');
document.getElementById('breakfastMenuItems').innerHTML = breakfastMenuItemsHTML;
```

- O código fornecido converte os itens do array do menu de café da manhã em strings HTML usando `map()` e uma função arrow para estruturar o formato HTML de cada item.
- Subsequentemente, um método de concatenação de strings foi usado para juntar as strings HTML geradas em uma única string coesa usando `join("")` para se preparar para inserção.

- Finalmente, este HTML concatenado é atualizado dinamicamente dentro do elemento HTML específico identificado pelo ID breakfastMenuItems com a string HTML concatenada, populando dinamicamente a página da web com os itens do menu de café da manhã em parágrafos formatados <p>.

Passo 3: Definindo o método de array forEach e a iteração do loop for

- Vamos percorrer o array **mainCourseMenu** usando o método de array **forEach**. Para isso, inclua o código abaixo no arquivo **restaurant_menu.js** após o código JavaScript anterior.

```
let mainCourseItem = '';
mainCourseMenu.forEach((item, index) => {
  mainCourseItem += `<p>Item ${index + 1}: ${item}</p>`);
document.getElementById('maincourseMenuItems').innerHTML = mainCourseItem;
```

- A variável mainCourseItem é inicializada como uma string vazia (""). Esta variável será usada para acumular strings HTML geradas para cada item do menu principal.
- O método forEach percorre cada elemento no array mainCourseMenu. Para cada item no array, a função de seta (item, index) => {...} é executada. Dentro da função de seta, uma string HTML é composta para cada item do menu, incorporando o conteúdo do item e seu índice correspondente.
- Em seguida, o conteúdo HTML é atualizado dinamicamente definindo a propriedade innerHTML do elemento HTML com o ID maincourseMenuItems para a string acumulada mainCourseItem, que irá inserir os itens do menu principal no elemento específico dentro da página da web.

- Por fim, vamos iterar o array **dessertMenu** usando a iteração do **loop for**. Para isso, inclua o código abaixo no arquivo **restaurant_menu.js** após o código JavaScript anterior.

```
let dessertItem = '';
for (let i = 0; i < dessertMenu.length; i++) {
  dessertItem += `<p>Item ${i + 1}: ${dessertMenu[i]}</p>`;
document.getElementById('dessertMenuItems').innerHTML = dessertItem;
```

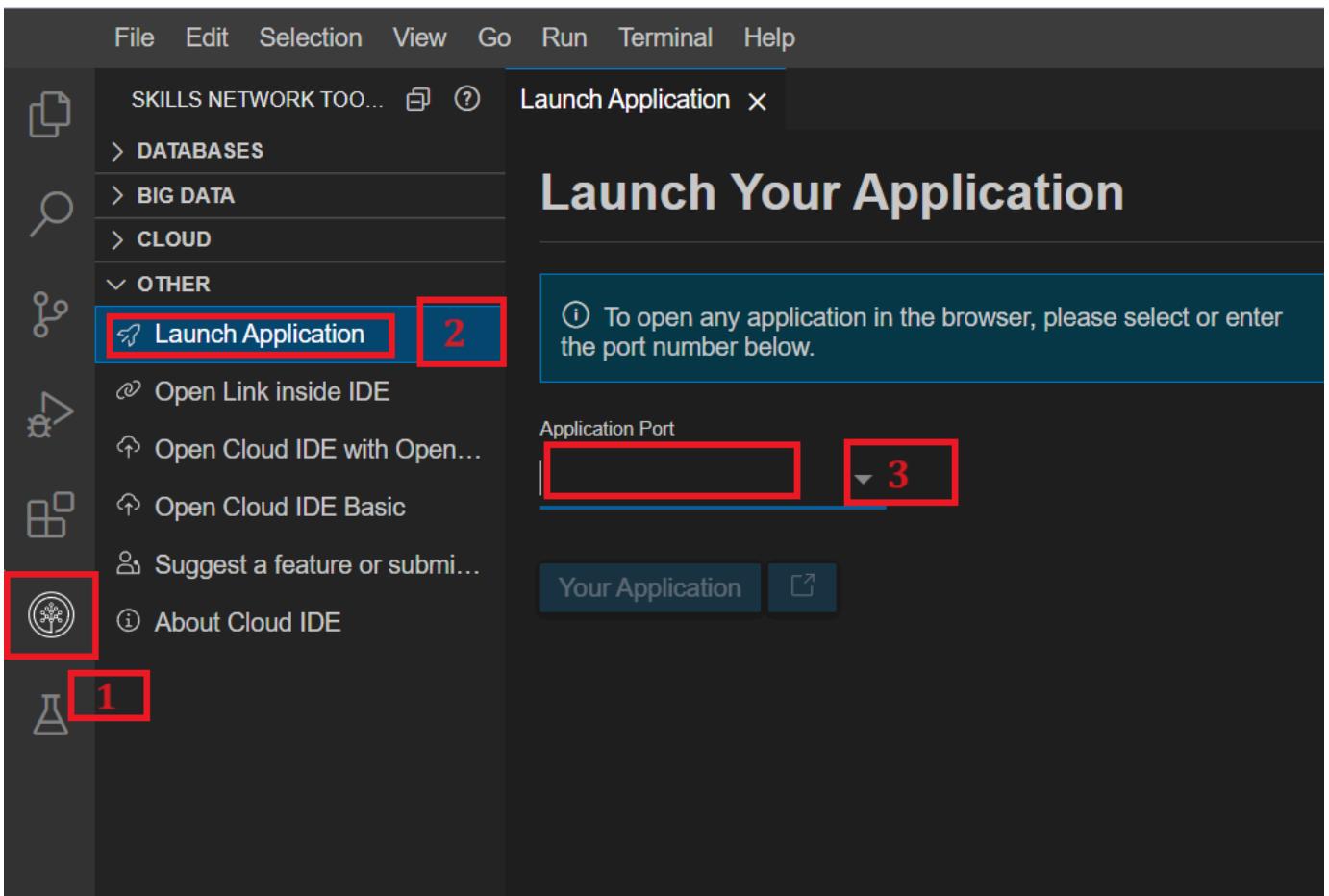
- A variável dessertItem é inicializada como uma string vazia (""). Esta variável serve como um recipiente para acumular strings HTML geradas para cada item do menu de sobremesa.
- O loop for itera pelo array dessertMenu, começando do índice 0 e continuando até alcançar o comprimento do array. Para cada iteração, uma string HTML é criada usando o item atual no array dessertMenu, incluindo o conteúdo do item e seu índice (i + 1).
- O conteúdo HTML gerado, composto por parágrafos individuais <p> contendo itens do menu de sobremesa com seus respectivos índices, é atribuído ao HTML interno do elemento HTML identificado pelo ID dessertMenuItems, que irá inserir os itens do menu de sobremesa no elemento específico dentro da página da web.

Passo 4: Verifique a saída

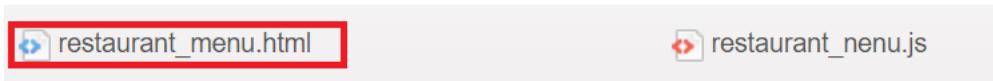
- Para visualizar sua página HTML, clique com o botão direito no arquivo **restaurant_menu.html** após selecioná-lo, e então selecione “Abrir com Live Server.”
- O servidor deve iniciar na porta 5500, indicado por uma notificação no canto inferior direito.



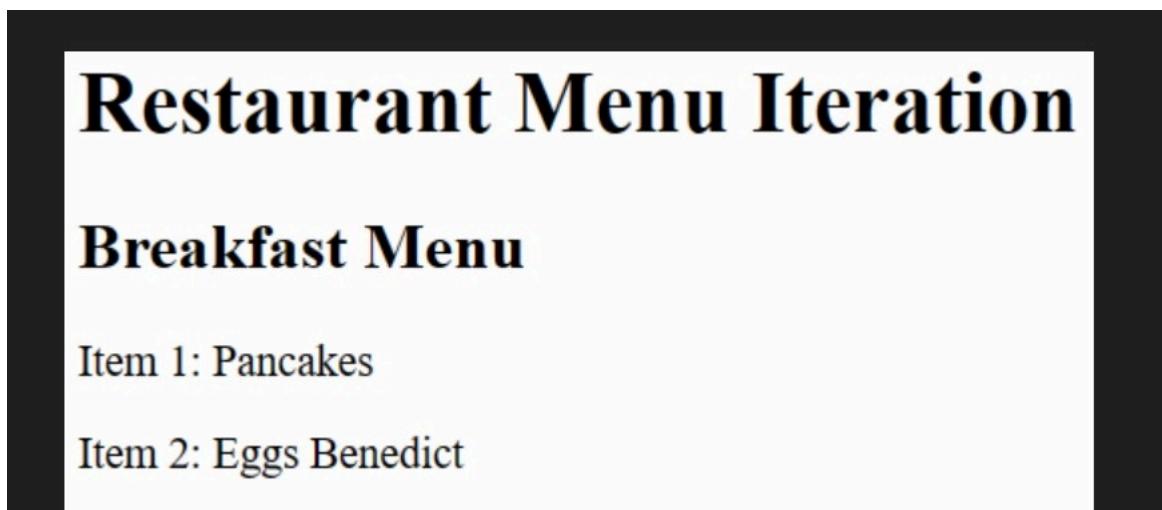
- Clique no botão Skills Network à esquerda (veja o número 1), isso abrirá a “Caixa de Ferramentas do Skills Network”. Em seguida, clique em **Iniciar Aplicativo** (veja o número 2). A partir daí, insira o número da porta como 5500 no número 3 e clique neste botão .



4. Isso abrirá seu navegador padrão onde você verá o nome da pasta **cloned-folder-name**. Clique no nome da pasta **cloned-folder-name** e depois clique na pasta **restaurantMenu**. Você verá arquivos relacionados a essa pasta onde você clicará novamente no arquivo **restaurant_menu.html** conforme mostrado abaixo.



5. Isso abrirá a página HTML e você verá os itens do menu para **Café da Manhã**, **Prato Principal** e **Sobremesa**.



Passo 5: Execute comandos Git

1. Execute `git add` para adicionar os arquivos e pastas mais recentes, escrevendo o comando fornecido no terminal no ambiente Git.

```
git add --a
```

Certifique-se de que o terminal tenha o caminho conforme abaixo:

1

```
theia@theia-richaar:/home/project/jscourseTesting$ git add --a
Replace jscourseTesting with your cloned repository folder name
```

2. Em seguida, execute `git commit` no terminal. Ao realizar `git commit`, o terminal pode mostrar uma mensagem para configurar seu `git config --global user.name` e `user.email`. Se sim, você precisará executar o comando `git config` também para `user.name` e `user.email`, conforme dado.

```
git config --global user.email "you@example.com"
```

```
git config --global user.name "Seu Nome"
```

Nota: Substitua os dados entre aspas pelos seus próprios detalhes.

Em seguida, execute o comando de commit conforme dado:

```
git commit -m "mensagem"
```

3. Depois, execute `git push` apenas escrevendo o comando fornecido no terminal.

```
git push origin
```

- Após o comando de push, o sistema solicitará que você insira seu nome de usuário e senha. Insira o nome de usuário da sua conta GitHub e a senha que você criou no primeiro laboratório. Após inserir as credenciais, todas as suas pastas e arquivos mais recentes serão enviados para seu repositório GitHub.

Nota: Após colar o código, salve seu arquivo. Você pode usar qualquer método de saída para isso. Se você editar seu código, simplesmente atualize seu navegador rodando pela porta 5500. Dessa forma, não há necessidade de lançar a aplicação repetidamente.

Tarefa de prática

1. Você pode incluir os preços de todos os itens no `breakfastMenu`, `mainCourseMenu` e `dessertMenu`.

2. Para isso, você pode incluir os preços da seguinte forma:

```
const breakfastMenu = ['Pancakes- $12', 'Eggs Benedict -$22.99', 'Oatmeal -$21.99', 'Frittata -$15'];
```

3. Em seguida, acesse-o de maneira semelhante e você verá a saída para os itens do menu junto com os preços também.

Restaurant Menu Iteration

Breakfast Menu

Item 1: Pancakes- \$12

Item 2: Eggs Benedict -\$22.99

Item 3: Oatmeal -\$21.99

Item 4: Frittata -\$15

Resumo

1. **Estrutura HTML:** Define uma estrutura HTML para um menu de restaurante, categorizando os itens em seções de Café da Manhã, Prato Principal e Sobremesa, utilizando cabeçalhos `<h2>` apropriados e elementos `<div>` com IDs únicos para exibição de conteúdo.
2. **Conteúdo dinâmico em JavaScript:** Utiliza JavaScript para preencher dinamicamente os itens do menu dentro de suas respectivas seções.
 - o Menu de café da manhã: Usa o método `map` para gerar strings HTML para os itens de café da manhã e as insere no elemento `<div>` designado.
 - o Menu de prato principal: Utiliza `forEach` para construir strings HTML para os itens do prato principal, atualizando o `<div>` relacionado com o conteúdo gerado.
 - o Menu de sobremesa: Utiliza um loop `for` para criar strings HTML para os itens de sobremesa e as insere no elemento `<div>` designado para os itens de sobremesa.
3. **Manipulação de dados:** Armazena os dados dos itens do menu em arrays JavaScript separados (`breakfastMenu`, `mainCourseMenu`, `dessertMenu`) para facilitar a criação de conteúdo dinâmico sem codificação rígida dentro da estrutura HTML.
4. **Renderização dinâmica:** Ilustra como o JavaScript interage com o HTML, gerando e atualizando dinamicamente o conteúdo dentro da página da web, demonstrando diferentes métodos de iteração para apresentar itens do menu sob categorias específicas.

© IBM Corporation. Todos os direitos reservados.