

Bootcamp IGTI: Desenvolvedor React

Trabalho Prático

Módulo 1: JavaScript Avançado

Objetivos de Ensino

Exercitar os seguintes conceitos vistos em sala de aula:

- ✓ Implementação de aplicação com JavaScript puro.
- ✓ Manipulação de Eventos.
- ✓ Requisições assíncronas.

Enunciado

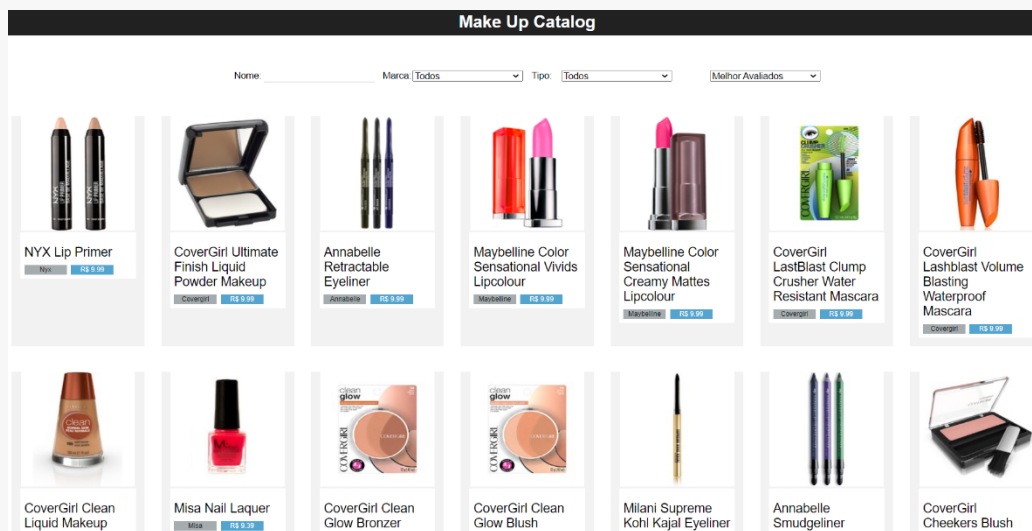
Construção de uma aplicação de catálogo de produtos de maquiagem a ser consumido de uma API utilizando JavaScript puro e HTML.

Atividades

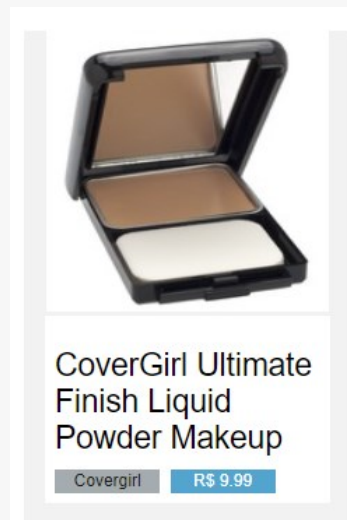
Os alunos deverão desempenhar as seguintes atividades:

1. Implementar em JavaScript puro, HTML e CSS uma aplicação para apresentação do catálogo de produtos de maquiagem. A URL base para consumo da API com os dados dos produtos é <https://makeup-api.herokuapp.com/>. As respectivas rotas serão descritas nas atividades subsequentes:
2. A imagem abaixo ilustra um exemplo de implementação desta aplicação, com um tipo de estilização. O estilo da página não necessariamente precisa ter o mesmo layout apresentado, mas é importante ter os mesmos componentes. O

projeto base (HTML e CSS) será disponibilizado no fórum de avisos do professor, caso queiram utilizá-lo como referência.



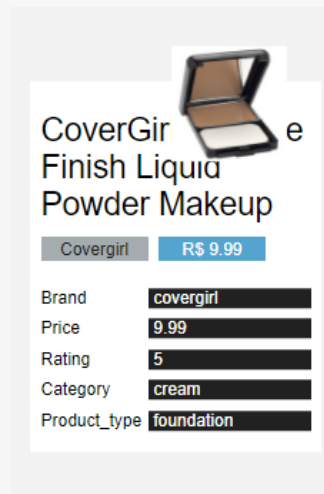
- No exemplo acima, foi utilizada somente a rota <http://makeup-api.herokuapp.com/api/v1/products.json> para carregar todas as informações necessárias para a aplicação, entretanto, existem rotas específicas disponíveis para carregar cada tipo de informação.
- A aplicação deverá possuir 3 filtros, sendo eles **Nome**, **Marca** e **Tipo** relacionados aos campos `name`, `brand` e `product_type`, respectivamente. Além dos filtros, será necessário incluir a ordenação dos produtos na página por **Melhor Avaliados** considerando o campo `rating`, por **Menores Preços** e **Maiores Preços** considerando o campo `price`, e por **A-Z** e **Z-A** relacionados ao campo `name`.
- Na apresentação para cada um dos produtos deve ser apresentada sua imagem, o nome do produto abaixo da imagem, a marca e o preço, conforme exemplo abaixo:



Segue um trecho de código HTML com o exemplo para o item apresentado.

```
<div class="product" data-name="CoverGirl Ultimate Finish Liquid Powder Makeup"
  data-brand="covergirl" data-type="foundation" tabindex="384" style="display: block;">
  <figure class="product-figure">
    
    </figure>
    <section class="product-description">
      <h1 class="product-name">CoverGirl Ultimate Finish Liquid Powder Makeup </h1>
      <div class="product-brands">
        <span class="product-brand background-brand">Covergirl</span>
        <span class="product-brand background-price">R$ 9.99</span>
      </div>
    </section>
    ....
  </div>
```

6. Para o preço do produto deverá ser aplicado um fator de conversão de R\$5,50 e exibido o seu preço com 2 casas decimais. Exemplo: Um price de 10.49 aplicado a um fator de 5,50, o produto apresentará o valor de R\$57,70.
7. Ao clicar no produto seus detalhes como Marca (brand), Preço (price), Avaliação (rating), Categoria (category) e Tipo (product_type) devem ser exibidos conforme imagem abaixo:



Segue um trecho de código HTML com o exemplo para o item apresentado.

```
<div class="product" data-name="CoverGirl Ultimate Finish Liquid Powder Makeup"
  data-brand="covergirl" data-type="foundation" tabindex="384" style="display: block;">
  ....
  <section class="product-details"><div class="details-row">
    <div>Brand</div>
    <div class="details-bar">
      <div class="details-bar-bg" style="width= 250">covergirl</div>
    </div>
  </div><div class="details-row">
    <div>Price</div>
    <div class="details-bar">
      <div class="details-bar-bg" style="width= 250">9.99</div>
    </div>
  </div><div class="details-row">
    <div>Rating</div>
    <div class="details-bar">
      <div class="details-bar-bg" style="width= 250">5</div>
    </div>
  </div><div class="details-row">
    <div>Category</div>
    <div class="details-bar">
      <div class="details-bar-bg" style="width= 250">cream</div>
    </div>
  </div><div class="details-row">
    <div>Product_type</div>
    <div class="details-bar">
      <div class="details-bar-bg" style="width= 250">foundation</div>
    </div>
  </div></section>
</div>
```

8. A página inicial deverá carregar todos os produtos considerando a ordenação por **Melhor Avaliados**, não sendo necessário realizar nenhuma paginação e nenhum filtro.

Dicas e sugestões:

- Utilize o projeto-base fornecido pelo professor como referência.

- Utilizem o fetch para consumo da API e os conceitos apresentados nas videosaulas para implementação do trabalho.
- Algumas informações dos produtos estão como null, portanto **para estes casos** será necessário desconsiderar a exibição da informação nos detalhes dos produtos bem como nos critérios para ordenações e filtros. **Exemplo:** Para informação numérica (rating, price, ...) considerar o valor 0 como default e para informações textuais (brand, product_type,...) considerar o valor "" como default.
- Alguns produtos não terão sua imagem carregada, utilizem a imagem default disponibilizada no projeto base para caso de erro no carregamento da imagem.
- Caso tenham dificuldades com a API, poderá ser utilizado como alternativa o json-server com o .json dos dados disponibilizado no projeto base.
- No questionário serão feitas perguntas sobre o JavaScript e sobre o projeto implementado.
- Caso alguma questão esteja incorreta, solicitarei ao IGTI que anule a mesma. Esse processo **não é "instantâneo" e leva algum tempo para acontecer – geralmente ao final do módulo.** Quando acontecer, **todos os alunos ganharão a pontuação da questão anulada.**
- Demonstrarei minha implementação na Primeira Aula Interativa.