

REVISÃO REACT

Responda o questionário abaixo, isso te fará revisar os assuntos principais de todo o conteúdo que vimos nas últimas semanas de aula.

Pode perguntar para a professora, monitores e monitoras, amigas da turma, consultar o material de resumo no repositório ou mesmo pesquisar no Google pois que importa é encontrar e entender as respostas!

1) Hoje em dia empresas trabalham com componentização de telas de site, o que seria **componentizar** um site? Quais são as **2 dicas** para saber se uma pedaço de tela é ou não um componente?

2) O React é uma biblioteca usada para criação de componentes de tela com Javascript, ele usa o conceito de **DOM Virtual**, o que é isso e qual a diferença entre **DOM virtual** e **DOM Real**?

3) Na figura abaixo criei um component **Link** usando **Javascript Puro**, como seria fazer isso usando **React.createElement**?

```
// <a class="link" href="/conta">Criar uma conta</a>
```

```
const linkCriarUmaConta = document.createElement('a')
linkCriarUmaConta.className = 'link'
linkCriarUmaConta.href = '/conta'
linkCriarUmaConta.appendChild(document.createTextNode('Criar uma conta'))
```

4) React faz a gente programar declarativamente, qual a diferença entre programação declarativa e imperativa?

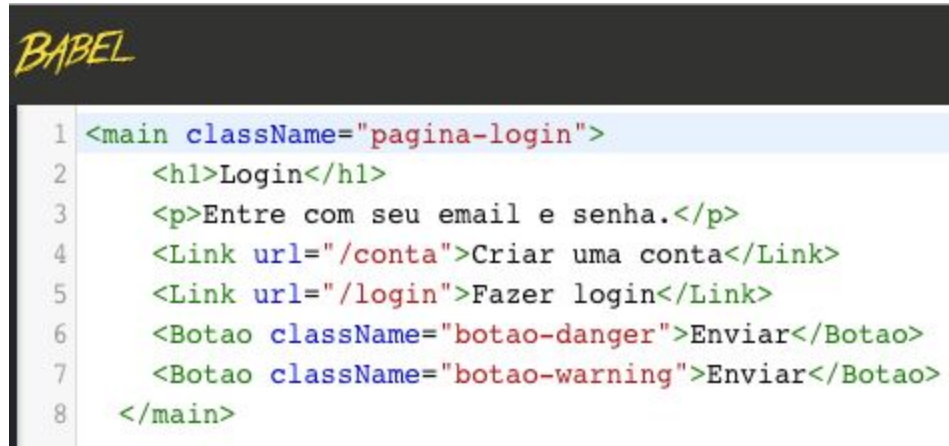
5) Além das vantagens já citadas, você consegue elencar pelo menos mais uma vantagem de usar React?

6) É obrigatório usar React sempre junto com Babel? O que o Babel faz?

7) A maioria de nós quando bate o olho em um código escrito com JSX entende o que aparecerá na tela porque se assemelha à HTML:

- O que a sigla JSX significa?

- As linhas de código na figura abaixo estão escritas em JSX, como ficará o código depois que o Babel passar por ele (depois que o Babel transpilar ele, ou seja, converter para uma outra sintaxe)?



```

1 <main className="pagina-login">
2   <h1>Login</h1>
3   <p>Entre com seu email e senha.</p>
4   <Link url="/conta">Criar uma conta</Link>
5   <Link url="/login">Fazer login</Link>
6   <Botao className="botao-danger">Enviar</Botao>
7   <Botao className="botao-warning">Enviar</Botao>
8 </main>

```

- Quais são os dois atributos que são diferente de HTML quando usamos em uma tag escrita com JSX?
- 8) Um componente é uma tag criada pela gente, quais são as duas formas de criar um componente?
- 9) Se tenho um componente `<Navbar />` e quero passar para a tag dele uma variável chamada **usuario**, como faço isso?
- 10) Em uma página uso a tag `<Navbar usuario={usuario} logaUsuario={logaUsuario} />` como acessar esses atributos/propriedades da tag dentro do código dela se o componente **Navbar** for uma função? E quando ele for classe, como que faço para usar esse usuário dentro do render ou chamar a função logaUsuario em um outro método dessa mesma classe?
- 11) Olhando para as tags abaixo como ficará o objeto **props** destes componentes?
- `<Login />`
 - `<Botao disabled>Enviar</Botao>`
 - `<Campo`
 `type="password"`
 `required`
 `minLength={6}`
 `onChange={habilitaOuDesabilita}`

/>

- 12) Porque chamamos a função **onChange** da tag **mãe** dentro da tag **filha**? Para te ajudar a responder, lembre-se da ordem de propagação de eventos, no caso o evento **change** é propagado de baixo para cima, ou seja, acontece primeiro na tag filha (veja o print abaixo):

```
valida = (evento) => {
  this.input = evento.target

  const regex = /^(([^<>()[]\.\.,;:\s@"]+(\.[^<>()[]\.\.,;:\s@"]+)*|"(.*")|'(.*)')@([0-9]{1,3}
  let mensagem = ''

  if (this.props.required && this.input.value.trim() === '') {
    mensagem = 'Campo obrigatório'
  } else if (this.props.minLength && this.input.value.length < this.props.minLength) {
    mensagem = `Digite pelo menos ${this.props.minLength} caracteres`
  } else if (this.props.type === 'email' && !regex.test(this.input.value)) {
    mensagem = 'Valor inválido'
  }

  this.setState({ erro: mensagem }, this.props.onChange)
}
```

- 13) Podemos passar o nome de duas funções para serem chamadas pelo navegador (Chrome, Firefox e etc) quando um evento acontece? Se não podemos qual a solução para quando um usuário clicar em uma opção do menu eu queira deslogá-lo e fechar o menu caso este esteja aberto?
- 14) Como eu faria um **"IF"** no meio do HTML de uma tag **Navbar** caso eu queria mostrar o nome do usuário se ele estiver logado e caso contrário mostrar o texto "Olá estranho"? Considere que o usuário foi passado dentro do **props** recebido pela função do meu componente **Navbar**.
- 15) O que é a palavra chave **"this"** que usamos dentro de **classes**? Qual a diferença entre classe e objeto/instância? Dê exemplos...
- 16) O que é herança entre classes, qual a palavrinha mágica para herdar? Crie uma classe **Pessoa** com as propriedades **nome** e **sobrenome**, depois outra classe chamada **Paciente** com **peso** e **altura** e faça **Paciente** herdar de **Pessoa**.
- 17) O que é a função **super** que usamos no **construtor** de uma classe com herança?

18) Crie uma classe **Animal** com as propriedades **nome** e **raça** e a função **locomover**, então, crie uma segunda classe chamada **Ave** que deverá herdar de **Animal** e reescrever a função **locomover** logando no console o texto *"Eu me locomovo voando"*. Em seguida crie uma terceira classe chamada **Peixe**, que herde também de **Animal** e reescreva a função **locomover** logando no console o texto *"Eu me locomovo nadando"*. Por fim crie uma última classe com nome de **Cobra**, ela deve herdar de **Animal** assim como **Peixe** e sobrescrever o método **locomover** com o log do texto *"Eu me locomovo rastejando"*.

19) Quando tenho que usar classe obrigatoriamente para criar um componente?

20) Em um componente que é classe qual desses métodos executa primeiro (vale recordar que métodos são funções que criamos dentro de uma classe)?

- a) render
- b) constructor
- c) handleClick

21) Em qual função de classe devemos inicializar o estado de uma tela? Como eu iniciaria o estado de um componente **Menu**, sendo que o estado inicial dele é fechado?

22) Uma tela só atualiza quando o estado dela muda, qual das alternativas abaixo é a maneira correta de atualizar o estado da tela para menu aberto?

- a) `this.state = { aberto: true }`
- b) `this.setState({ aberto: true })`
- c) `this.state = new State({ aberto: true })`

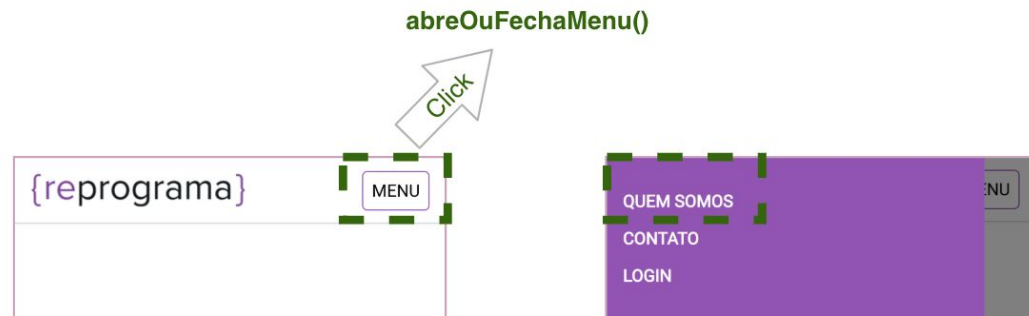
23) O método **setState** faz duas coisas que coisas são essas?

24) Sobre ref, responda as perguntas abaixo:

- Quando é uma boa ideia usar ref?
- Como criar uma referência e onde ela deve ser criada?
- Como vincular uma referência à uma tag?
- Como acesso o valor corrente guardado dentro de uma referência?

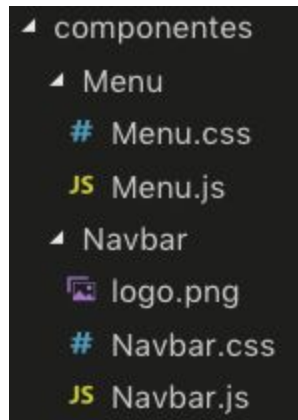
25) Respondendo as perguntinhas abaixo crie o componente Menu:

- 1) O componente pode mudar de estado? **Sim // class**
 - 2) O que muda? **setState({ aberto: true })** ou **setState({ aberto: false })**
 - 3) Qual o estado inicial? **this.state = { aberto: false }**
 - 4) O que faz ele mudar?
- É o evento de clique no botão menu e nas opções do menu // function onClick**



26) Geralmente nas empresas não escrevemos o código todo em um único arquivo, a gente organiza nosso código em pastas e arquivos separados. Para fazer isso usamos uma ferramenta chamada Webpack que nos permite exportar e importar arquivos como js, css e imagens. Responda:

- O que acontece se não exporto um código e tento usá-lo em outro arquivo?
- Como ficaria a linha de código para exportar por exemplo uma função de componente chamada **Menu**?
- Como eu faria para importar um **Menu** para usá-lo em outro arquivo chamado **Navbar**, supondo que cada tem a sua própria pasta, ambas dentro de uma pasta chamada “componentes” e o **Menu** foi exportado como default? Figura abaixo:



- Se eu exportei o **Menu** sem usar a palavrinha **default**, como devo importar ele no **Navbar**?
- Imagina que quero dar uma apelido (um alias) para poder importar um componente **Link** com um nome diferente, no caso, invés de usar **Link** quero usar o nome **RouterLink**, como importar usando este apelido?
- Se quero importar um arquivo que eu mesma criei tenho de informar qual o caminho de onde o arquivo está para que o Webpack importá-lo, mas e quando eu quero importar um arquivo que está na pasta **node_modules**, por exemplo, como seria o comando para importar o **React** de dentro da pasta **node_modules**?

27) Falando em **node_modules**, durante as aulas paramos de baixar manualmente as bibliotecas que usamos (tais como *react*, *react-dom*, *react-router-dom*, *react-scripts* entre outras) porque estamos usando o npm (gerenciador de pacotes do node) para ele baixar e salvar os arquivos pra gente dentro de uma pasta chamada **node_modules** (lembrando que o Node chama arquivos de módulos, daí que vem esse nome de pasta, ou seja, módulos do node ou "node_modules"). Responda:

- O que é o arquivo **package.json** e qual é o comando para criá-lo caso não exista?
- Qual é o comando para instalar uma nova dependência? Chamamos de dependência toda biblioteca que o nosso projeto usa.

- O que acontece se eu não uso a flag "--save" quando rodo o comando de instalação de dependência?
- Nós usamos o **react-scripts** para criar comandos nossos tais como "start" e "build", em que local do arquivo **package.json** criamos comandos?

28)Quanta coisa aprendemos, e outra delas foi como hospedar um site criado com React no **Netlify**, qual é o passo-a-passo para fazer deploy, ou seja, "por no ar" uma nova versão de um site?

Dica: não se esqueça de rodar o comando "*npm run build*" para que o react-scripts gere a pastinha com o código otimizado prontinho para subir ;-)

29)Nos usamos várias tags do **React Router DOM**, qual a diferença entre as tags **<Link />** e **<NavLink />**

30)Como mapear uma url **"/conta"** para um componente chamado **Conta**?

31)Como faço um **Switch** de rotas para que caso nenhuma das rotas acima deem match com as url digitada pelo usuário eu então exiba um componente de página não encontrada.

32)Pratique você mesma a **leitura da documentação do React Router** para descobrir como configuramos ele em um site acessado por um Browser. Sempre que usamos uma biblioteca de terceiros a documentação é nossa melhor amiga pois explica como a biblioteca deve ser instalada e usada :-)