Atividade 01

Desenvolvimento de Software para WEB
Projeto de Interface Web

Conteúdo: Passagem de props, múltiplos componentes, props.children e React.Children

Obs.: Faça os exercícios abaixo usando o CodeSandBox ou o ambiente de trabalho instalado em sua máquina (VSCode + Node, por exemplo).

01 - Siga as instruções:

- Crie dois arquivos dentro da pasta src/components/atividade01/questao01 chamados 01Pai.jsx e 01Filho.jsx
- Dentro de 01Pai.jsx, importe o componente Filho do arquivo 01Filho.jsx
- No componente Pai faça uma chamada do componente Filho da seguinte forma:
    - …
    - <Filho altura={1.8} peso={90} />
- O componente Filho em 01Filho.jsx deverá calcular o imc = peso/(altura*altura) e exibir o resultado
- Além disso, em Filho, você deverá criar uma função interna que retorna uma mensagem JSX no seguinte casos:
    - imc < 18, exibir <h3>Abaixo do peso</h3>
    - imc > 25, exibir <h3>Acima do peso</h3>
    - <h3>Peso ideal</h3> para os outros casos.
- Resumindo, o componente Pai deverá passar, via props o peso e altura para Filho. Este deve calcular o IMC e exibir as mensagens (em JSX) de acordo com o cálculo do resultado.

02 - Siga as instruções:

- Crie um arquivo dentro da pasta src/components/atividade01 chamado 02MeuPC.jsx.
- Dentro do arquivo, crie os componentes internos PlacaMae, Memória e PlacaDeVideo
- Cada componente deve receber, via props, um nome e um preço e o exibir em seu JSX
- Exporte os componentes usando o export da forma que quiser
- Em App.js importe todos os componentes usando um alias “PC”. Chame os componentes no JSX de App.js

03 - Siga as instruções:

- Crie um arquivo dentro da pasta src/components/atividade01 chamado 03Batalha.jsx
- Nele, crie os seguintes componentes internos, usando funções:
- “Hero”, o qual imprimirá a propriedade “name”, recebida via “props”. Além disso, mostre uma imagem de sua escolha, usando a tag <img>”.
- “Enemy”, a mesma coisa de Hero, com um “name” e uma imagem.
- “Arena”. Irá renderizar os componentes “Hero” e “Enemy”

    Ex. (JSX de Arena):
    ```
    <div>
        <Hero name = “Baki” img = “../baki.png”/>
        <Enemy name = “Yujiro” img = “../ogre.png”/>
    </div>
    ```

- “World”, o qual terá em seu jsx uma chamada ao props.children.

- Exiba World em App.js da seguinte forma:

```
<World>
    <Arena />
    <Arena />
    <Arena />
</World>
```

OPCIONAL: Modifique o componente Arena para que ele passe a trabalhar com React.Children. Além disso, Arena terá um props chamado “arena”, o qual conterá o seu nome. Esse props deve ser repassado para suas tags filhas e exibidas nelas, usando o cloneElement. Ex.:

```
<Arena arena = “Tokio Dome – Underground Arena” >
    <Hero name="Sicrano de Tal" img="hero.jpg">
    <Enemy name="Fulano de Tal" img="enemy.png">
</Arena>
```

No caso acima, Hero e Enemy devem exibir a informação referente ao nome da Arena.