SEGUNDA AVALIAÇÃO PARCIAL
SEGUNDA CHAMADA

DESENVOLVIMENTO DE SOFTWARE PARA WEB
PROJETO DE INTERFACES WEB

2023.1 - PROF. JEFFERSON DE CARVALHO SILVA

UFC - QUIXADÁ

Questão 01 (2.0pts)	Questão 02 (2.0pts)	Questão 03 (2.0pts)	Questão 04 (2.0pts)	Questão 05 (2.0pts)

Obs.:

Questão 01 - Seja o vetor (ou array) `const vet = [4,9,16,64]`. Implemente as seguintes funções (apenas as funções) (0,5pt cada):

A. Implemente uma função **arrow** que receba como parâmetro `vet` e retorne um novo vetor onde cada elemento do novo vetor é um elemento de `vet` multiplicado por 2. No exemplo acima, a sua função deve retornar `[8,18,32,128]`. Use `Array.prototype.map()`.
B. Refaça a questão acima (A) em uma nova função **function** só que dessa vez você não pode usar `Array.prototype.map()`. Use um laço comum.
C. Implemente uma função **arrow** que receba como parâmetro `vet` e retorne um novo vetor onde cada elemento do novo vetor é a raiz quadrada de um elemento de `vet`. No exemplo acima, a sua função deve retornar `[2,3,4,8]`. Use `Array.prototype.map()` .
D. Refaça a questão acima (C) em uma nova função **function** só que dessa vez você não pode usar `Array.prototype.map()`. Use um laço comum. sd

Obs.: Em JavaScript, a raiz quadrada é calculada com `Math.sqrt(x)` onde `x` é o número cuja raiz quadrada deve ser calculada.

Questão 02 - Seja o código abaixo:

```
<>
  <Universidade sigla = "UFC" cidade = "Quixadá">
    <Curso nome = "Design Digital" />
    <Curso nome = "Sistemas de Informação" />
    <Curso nome = "Engenharia de Software" />
  </Universidade>
</>
```

Implemente apenas o **return** do Componente `Universidade` onde você deverá usar `Children.map` com o objetivo de repassar para os componentes filhos as props `sigla` e `cidade`. **ATENÇÃO**: caso o curso seja "Engenharia de Software", você não deve repassar a props `cidade`, apenas a `sigla`.

Questão 03 - Crie um componente em React que liste um array de objetos alunos, por exemplo:

```
const [alunos, setAlunos] = useState([
    {nome:"AAAA",np1:2,np2:5},
    {nome:"BBBB",np1:6,np2:5.5},
    {nome:"CCCC",np1:2,np2:8.9},
    {nome:"DDDD",np1:10,np2:5},
    {nome:"EEEE",np1:10,np2:10}
])
```

No return deste componente, você deve usar um **Map** para renderizar os elementos do Array em uma table comum do HTML (não precisa usar CSS ou MUI). Para CADA aluno, você também deve incluir a sua RESPECTIVA média (crie uma célula ao lado de np2 para mostrar a média calculada, (np1+np2)/2).

Mude a cor do nome de cada aluno que ficar com a média abaixo de 5 para VERMELHO (use CSS).

Questão 04 - Faça o que se pede:

- Implemente uma "Promisse" que, após 5 segundos, sorteia um número randômico. Caso o número seja ímpar, a promessa "resolve" um objeto JSON com seu nome e curso. Caso contrário, a promessa "**reject**" uma mensagem de "ERRO!". (1,0pt)
- Implemente uma função assíncrona que "**await**" o resultado da promessa do item anterior e exiba seu resultado, em caso de resolve, em console.log. A função também deve exibir o erro caso a promessa seja rejeitada (try-catch) (1,0pt).

Questão 05 - Refaça a Questão 04 usando o **then** e o **catch** para o tratamento da promessa (1,5pt). Qual a diferença entre as abordagens await e the-catch (0,5pt)?