

Bootcamp: Desenvolvimento Full Stack

Desafio do módulo

Módulo 3	FrontEnd com React
----------	--------------------

Objetivos

Exercitar os seguintes conceitos trabalhados no Módulo:

- ✓ Implementação de algoritmos com JavaScript.
- ✓ Criação de componentes com React.
- ✓ Utilização de class components, functional components ou hooks.

Enunciado

Criar uma aplicação com React, realizar cálculo de valorização/depreciação com base em um montante, taxa de juros mensal e quantidade de parcelas mensal, utilizando o conceito de juros compostos.

Atividades

Os alunos deverão desempenhar as seguintes atividades:

1. Criar o projeto com o pacote **create-react-app** ou utilizar o projeto **_react-projeto-base** já disponibilizado pelo professor no fórum do módulo. Essa última é a forma recomendada pelo professor.
2. Definir os elementos que farão parte do estado da aplicação. O estado da aplicação deve ser composto do **valor inicial**, a **taxa de juros mensal** e o **período**. A taxa de juros pode ser tanto positiva (investimento) quanto negativa (depreciação). A implementação da manipulação do estado da aplicação pode ser

feita tanto com **class components** quanto **hooks**, ficando a escolha a critério do aluno. Entretanto, as questões do desafio serão elaboradas com base em conceitos e implementação referentes somente aos React Hooks.

- O aluno deve pesquisar/investigar como deve ser feito o cálculo das prestações com base nas imagens abaixo e com base no vídeo de apresentação do professor. Há várias formas de se implementar esse cálculo.
- A interface fica a critério do aluno. O foco da avaliação serão os cálculos e conceitos relacionados ao React Hooks.
- As imagens abaixo podem servir de orientação para o aluno.

React - Juros Compostos

Montante inicial:
1000

Taxa de juros mensal:
0.5

Período (meses):
1

R\$ 1.005,00
1 +R\$ 5,00
0,50%

Tela inicial da aplicação, que indica que **R\$ 1.000,00** aplicados a uma taxa de juros de **0,5% ao mês**, **rende** aproximadamente **0,50%** ou **R\$ 5,00** após **1 mês**.

React - Juros Compostos

Montante inicial:
5900

Taxa de juros mensal:
0.8

Período (meses):
12

R\$ 5.947,20
1 +R\$ 47,20
0,80%

R\$ 5.994,78
2 +R\$ 94,78
1,61%

R\$ 6.042,74
3 +R\$ 142,74
2,42%

R\$ 6.091,08
4 +R\$ 191,08
3,24%

R\$ 6.139,81
5 +R\$ 239,81
4,06%

R\$ 6.188,92
6 +R\$ 288,92
4,90%

R\$ 6.238,44
7 +R\$ 338,44
5,74%

R\$ 6.288,34
8 +R\$ 388,34
6,58%

R\$ 6.338,65
9 +R\$ 438,65
7,43%

R\$ 6.389,36
10 +R\$ 489,36
8,29%

R\$ 6.440,47
11 +R\$ 540,47
9,16%

R\$ 6.492,00
12 +R\$ 592,00
10,03%

Tela após mudança de estado, que indica que **R\$ 5.900,00** aplicados a uma taxa de juros de **0,8% ao mês**, **rende** aproximadamente **10,03%** ou **R\$ 592,00** após **12 meses**.

React - Juros Compostos

Montante inicial:	Taxa de juros mensal:	Período (meses):
70000	-2	24

R\$ 68.600,00 1 -R\$ 1.400,00 -2,00%	R\$ 67.228,00 2 -R\$ 2.772,00 -3,96%	R\$ 65.883,44 3 -R\$ 4.116,56 -5,88%	R\$ 64.565,77 4 -R\$ 5.434,23 -7,76%	R\$ 63.274,46 5 -R\$ 6.725,54 -9,61%	R\$ 62.008,97 6 -R\$ 7.991,03 -11,42%
R\$ 60.768,79 7 -R\$ 9.231,21 -13,19%	R\$ 59.553,41 8 -R\$ 10.446,59 -14,92%	R\$ 58.362,34 9 -R\$ 11.637,66 -16,63%	R\$ 57.195,10 10 -R\$ 12.804,90 -18,29%	R\$ 56.051,19 11 -R\$ 13.948,81 -19,93%	R\$ 54.930,17 12 -R\$ 15.069,83 -21,53%
R\$ 53.831,57 13 -R\$ 16.168,43 -23,10%	R\$ 52.754,94 14 -R\$ 17.245,06 -24,64%	R\$ 51.699,84 15 -R\$ 18.300,16 -26,14%	R\$ 50.665,84 16 -R\$ 19.334,16 -27,62%	R\$ 49.652,52 17 -R\$ 20.347,48 -29,07%	R\$ 48.659,47 18 -R\$ 21.340,53 -30,49%
R\$ 47.686,28 19 -R\$ 22.313,72 -31,88%	R\$ 46.732,56 20 -R\$ 23.267,44 -33,24%	R\$ 45.797,91 21 -R\$ 24.202,09 -34,57%	R\$ 44.881,95 22 -R\$ 25.118,05 -35,88%	R\$ 43.984,31 23 -R\$ 26.015,69 -37,17%	R\$ 43.104,62 24 -R\$ 26.895,38 -38,42%

Tela após mudança de estado, que indica que **R\$ 70.000,00** com taxa de depreciação de **2% ao mês**, **desvaloriza** aproximadamente **38,42%** ou **R\$ 26.895,38** após **24 meses**.

Algumas dicas com base na implementação feita pelo professor, que foi feita com a utilização de React Hooks:

1. Faça com que os três inputs do estado da aplicação sejam do tipo **number** para conseguir manipular os dados com as setas “para cima” e “para baixo” do teclado.
2. Será cobrado nas questões apenas o valor do montante até 100.000 reais, taxa de juros entre -12% e 12% ao mês e parcelas de 1 a 36.
3. Além dos três inputs, crie mais uma variável de estado que será responsável por controlar as “parcelas”.
4. Faça a implementação do cálculo das parcelas com **useEffect**, utilizando como **deps** as variáveis de estado referentes aos três inputs. Nesta implementação, eu (Raphael Gomide) **não reaproveitei** o valor atual das parcelas.
5. Quebre a aplicação em quatro componentes: <App />, <Form /> (com os três inputs), <Installments /> e <Installment /> (parcelas/parcela, em inglês).
6. Não deixem de assistir o vídeo de apresentação deste desafio, onde demonstro a aplicação em funcionamento e dou mais algumas dicas.