

STEP Javascript - homework métodos e funções anônimas

Construir um tabuleiro de xadrez com as peças em suas posições iniciais (*em homeworks posteriores, aumentaremos as funcionalidades deste exercício*)

Requerido: usar arrays multidimensionais para o tabuleiro

Requerido: *tudo mais no código, incluindo as peças, será objeto, método e função anônima.*

Variáveis, se houverem, serão apenas variáveis de trabalho, temporárias. Este exercício é para você se exercitar principalmente nas funções anônimas. Tente usar funções anônimas antes mesmo de usar métodos em objetos (no esquema clássico de declarar “class”s)

Nota importante: *algumas das notações de arrays que vimos em sala de aula ficaram muito complicadas (porque exigem representações mais complexas que um array), portanto vamos utilizar a notação que vai mais abaixo. Como os arrays multidimensionais são concebidos como arrays dentro de arrays, o melhor é imaginar um array com oito posições (como se fosse uma linha do tabuleiro visualizada de forma vertical), e com um array de oito posições dentro de cada elemento deste array anterior, sendo que o array mais de dentro não tem nome:*

Isto equivale a:

```
meuArray = []      // um array unidimensional  
meuArray[0] = ["a", "b", "c"]    // um array unidimensional sem  
                                / nome dentro da posição zero de meuArray  
  
// e "a" estará em meuArray[0][0]  
// "b" estará em meuArray[0][1], etc
```

Se fizermos:

```
meuArray  
  
|__| contém : ["T", "C", "B", "R", "Q", "B", "C", "T"]  
  
|__| (...)  
  
|__|  
  
|__|  
  
(...)
```

Isso também significa que nesta primeira abordagem, não usaremos métodos como *push*, *pop*, *splice*, etc, mas referências diretas na forma matriz[[linha]][coluna].

Uma vez tendo conseguido isso internamente no javascript, tente então melhorar o visual no hmtl (note que a matriz acima apenas representa posições de peças umas em relação à outras: **isto é uma primeira abordagem para a separação completa entre o código “core” de processamento, e código mais simples que apenas faz visualização:** em outras palavras, o que vai ser materialmente exibido nada tem a ver com a representação interna no código), tente exibir os valores de cada elemento dentro de tags que você pega com getElementById, pode ser alguma <div> ou tags que você organizou para um visual melhor, como vários <p>s, <tl>s, <td>s, s etc. Talvez, inclusive, quem sabe você pode usar imagens em tags cuja imagem varia dependendo do valor do elemento na matriz!

=====

Entregue A PASTA (diretório) com os arquivos .html, .js e outros se houverem. Não entregue .txt, .docx, etc. Faça um zip da pasta e upload deste zip

=====