

```
//FLATMAP():  
//O "flatMap()" - Mapeamento Achatado - é um método inexistente no Javascript, mas, que os desenvolvedores inventam quando desejam  
achatar uma matriz. O que seria achatar uma matriz? Seria criar uma forma de gerar um array - ou outro tipo de dado - que retorne  
para nós somente os valores desejamos de uma matriz.  
//O flatMap() geralmente é criado no próprio prototype do Array, e ele usa a função map() como método de mapear um array para gerar  
um resultado...
```

```
//USANDO FLATMAP() PARA PEGAR SOMENTE AS NOTAS DE 2 TURMAS:
```

```
const escola = [{ //Perceba que aqui temos um array, que contém 2 objetos, e dentro destes objetos temos um array e com mais 2  
objetos, mas queremos retirar daqui só as notas e colocá-las todas dentro de um único array...
```

```
  nome: 'Turma 1',  
  alunos: [{  
    nome: 'Gustavo',  
    nota: 8.1  
  }, {  
    nome: 'Ana',  
    nota: 9.3  
  }]  
}, {  
  nome: 'Turma 2',  
  alunos: [{  
    nome: 'Rebeca',  
    nota: 8.9  
  }, {  
    nome: 'Roberto',  
    nota: 7.3  
  }]  
}]
```

```
const getNota = aluno => aluno.nota //Perceba que criamos uma callback que irá chamar as notas do array de alunos...
```

```
const getTurma = turma => turma.alunos.map(getNota) //E criamos uma callback para iterar em escola, iterando á partir do objeto que  
representa a turma, entrando onde temos alunos e fazendo um map dentro de alunos para achar as notas...
```

```
const notas = escola.map(getTurma) //Como resultado as notas são retornadas...
```

```
console.log('Incorreto:', notas) //Porém, quando imprimimos as notas elas ainda estão separadas por matriz, para resolver isso  
podemos usar um flatMap()...
```

```
Array.prototype.flatMap = function(callback) { //Criamos uma função interna o objeto Array chamada flatMap()  
    return Array.prototype.concat.apply([], this.map(callback)) //No retorno tivermos que usar todo o caminho de Array.prototype para  
    alcançar o concat, pois lembre-se que você está criando um método dentro da função Array, para referenciar ele tem que fazer todo o  
    caminho. O apply está sendo usado como uma espécie de filtro para o concat(), pois o concat não aceita funções, e o que estamos  
    passando é justamente uma função, por isso o apply foi usado para que o map aconteça dentro dele, e ele repasse para o concat somente  
    os resultados do map. Outra coisa importante é o this antes do map(), o this faz com que o map() seja executado de acordo com a  
    variável chamadora, do contrário o map() não teria ninguém para chamá-lo e daria erro.  
}  
  
console.log('Correto:', escola.flatMap(getTurma))
```

RESULTADO NO CONSOLE...

```
[Running] node "c:\Users\User\Documents\JAVASCRIPT\ARQUIVOS_DAS_AULAS\114-flatMap()__Achatando_Matrizes.js"  
Incorreto: [ [ 8.1, 9.3 ], [ 8.9, 7.3 ] ]  
Correto: [ 8.1, 9.3, 8.9, 7.3 ]  
  
[Done] exited with code=0 in 0.167 seconds
```