

Übungsblatt Fibonacci-Folge

Erklärung

Die Fibonacci Zahlenfolge ist so definiert:

1. Die ersten beiden Zahlen lauten 0 und 1.
2. Jede weitere Zahl in der Folge wird durch Addition ihrer beiden Vorgänger gebildet.

Daher folgt (ganz links: indices des arrays):

```
2 | 0 + 1 => 1 => [ 0, 1, 1 ]
3 | 1 + 1 => 2 => [ 0, 1, 1, 2 ]
4 | 1 + 2 => 3 => [ 0, 1, 1, 2, 3 ]
5 | 2 + 3 => 5 => [ 0, 1, 1, 2, 3, 5 ]
6 | 3 + 5 => 8 => [ 0, 1, 1, 2, 3, 5, 8 ]
7 | 5 + 8 => 13 => [ 0, 1, 1, 2, 3, 5, 8, 13 ]
```

Aufgabe 1: function getFib (n)


Rückgabewert der Funktion sei die **n-te** Fibonacci-Zahl, beginnend mit dem Index 0.

```
getFib(0) => 0
getFib(1) => 1
getFib(2) => 1
getFib(6) => 8
```

Implementiere diese Funktion!

Aufgabe 2: Klasse Fib

Baue jetzt die Funktion aus Aufgabe 1 in eine Klasse ein, die Klasse soll sich in einem Array die bereits gefundenen Fibonacci-Zahlen "merken". Jeder Aufruf von **getFib(n)** liefert entweder eine schon gefundene Zahl zurück, oder erweitert das interne Array entsprechend und gibt dann die richtige Zahl zurück!

 Fib
□ #fibArray: Number[]
● constructor()
● getFib(n): Number

Gutes Gelingen!