Day 16, Today's Topic
Destructuring Part 3

DAILY JS

```
/* ============================================================
## ================== Daily JS - Day 16 ==================
## ============================================================
*/
```

Day 16: Destructuring Part 3

Welcome back to the day 16, and the 3rd day of
discussing destructuring.

In the previous days, we discussed about how to
extract properties out of an object using destructuring
and how to apply destructuring in function arguments.

As you might have guessed, today we are going to discuss
about destructuring the elements out of an array.

Yes, you heard it right, not only can we extract the
properties out of objects, but we can also extract
elements out of arrays (and as we will see further,
using destructuring, we can extract out almost any element
even in the most complex and nested object).

```
/* ============= madhavbahl.tech/dailyjs/day16 ==============
## ======== Join Discord: madhavbahl.tech/discord-c2e ========
*/
```

```
/* ========================================================================
## ======================= Daily JS - Day 16 =======================
## ========================================================================
*/
```

Day 3. Pulling values out of arrays

So today's topic is going to be how to extract elements out of an array using destructuring instead of array elements.

Syntax

Let's first have a look at the syntax before taking the example

```js
  var myArr = [ 10, 20, 30, 40, 50 ];
  var [ ele1, ele2, ele3, ele4, ele5 ] = myArr;
```

ele1 to ele5 will be initialized with values according to the index of `myArr`. Alright, pretty easy, correct?

Now you must be thinking,
> Wait Madhav, what if I give more than `ele5` in while destructuring? There are only 5 elements, what will happen then?

Pretty genuine doubt, and I like it. Nothing will happen, t he value of excess variables will be `undefined`.

```
/* ================== madhavbahl.tech/dailyjs/day16 ==================
## ============== Join Discord: madhavbahl.tech/discord-c2e ==============
*/
```

```
/* ==========================================================================
## ============================== Daily JS - Day 16 ==========================
## ==========================================================================
*/
  ## Example 1

  **Given an array having 3 elements, which are the values of red, green and blue of
  an rgb color code. Extract those values into different variables.**

  Pretty easy, yes. Let's see how to do this without destructuring first.

  ```js
    var rgb = [ 254, 135, 97 ];
    var red = rgb[0];
    var green = rgb[1];
    var blue = rgb[2];
    console.log (`Red: ${red} | Green: ${green} | Blue: ${blue}`);
  ```


  Now again, you might be thinking, what's the problem with this?

  To be honest, there is no problem with this syntax.
  But then again, as we can see, we have to write 3 separate lines to extract
  out those elements and with destructuring, we can reduce it to a single line.

  ```js
    var rgb = [ 254, 135, 97 ];
    var [ red, green, blue ] = rgb;
    console.log (`Red: ${red} | Green: ${green} | Blue: ${blue}`);
  ```


  As you can see, this reduced the 3 lines where we extracted the elements into 1.

/* ========================= madhavbahl.tech/dailyjs/day16 ===================
==================== Join Discord: madhavbahl.tech/discord-c2e ============
*/
```

```
/* ==================================================================== */
/* ============================ Daily JS - Day 16 ===================== */
/* ==================================================================== */


  ## Example 2

  **Given an array of student names, extract out the elements into different variabes,
  then write a function which adds a given student as the first element**

  Let's do this without destructuring first.

  ```js
  var students = [
      'Casper',
      'Amanda',
      'Kepler'
  ];
  // Without destructuring
  var student1 = students[0];
  var student2 = students[1];
  var student3 = students[2];
  console.log (`
      Student 1: ${student1}
      Student 2: ${student2}
      Student 3: ${student3}
  `);
  function addStudent1 (studName, studArr) {
      // Note* this can be done using the `splice` method also,
      // But I'll do this the long way :)
      var newStudArr = [ studArr[0], studName ];
      for (let i=1; i<studArr.length; i++) {
          newStudArr.push (studArr[i]);
      }
      return newStudArr;
  }
  console.log ("New Student Array: ", addStudent1 ("John", students));
  ```


 As you can see, it's too much code. Let's see how it goes using features like
 rest and destructuring.

/* ============================ madhavbahl.tech/dailyjs/day16 ================= */
/* ========================= Join Discord: madhavbahl.tech/discord-c2e ======== */
```

```
/* ================================================================
## ======================== Daily JS - Day 16 =====================
## ================================================================
*/
  ## Solution to the student name problem using destructuring

  **Given an array of student names, extract out the elements into different
  variabes, then write a function which adds a given student as the first element**

  ```js
    var students1 = [
        'Casper',
        'Amanda',
        'Kepler'
    ];

    var [ stud1, stud2, stud3 ] = students1;

    console.log (`stud1: ${stud1} | stud2: ${stud2} | stud3: ${stud3}`);

    function addStudent2 (studName, studArr) {
        return [ studName, ...studArr ];
    }
    console.log ("New Student Array: ", addStudent2 ("Rocky", students1));
  ```

  So, that's it for today, I hope you understood the concept. See you tomorrow!

/* ======================= madhavbahl.tech/dailyjs/day16 =====================
================== Join Discord: madhavbahl.tech/discord-c2e =================
*/
```

THE
LEAN
PROGRAMMER | CodeTo Express

Thank you!

Feel free to reach out…

Email: theleanprogrammer@gmail.com

Web: madhavbahl.tech/

Github: github.com/MadhavBahlMD

LinkedIn: linkedin.com/in/madhavbahl/

Insta: instagram.com/theleanprogrammer/

DAILY JS