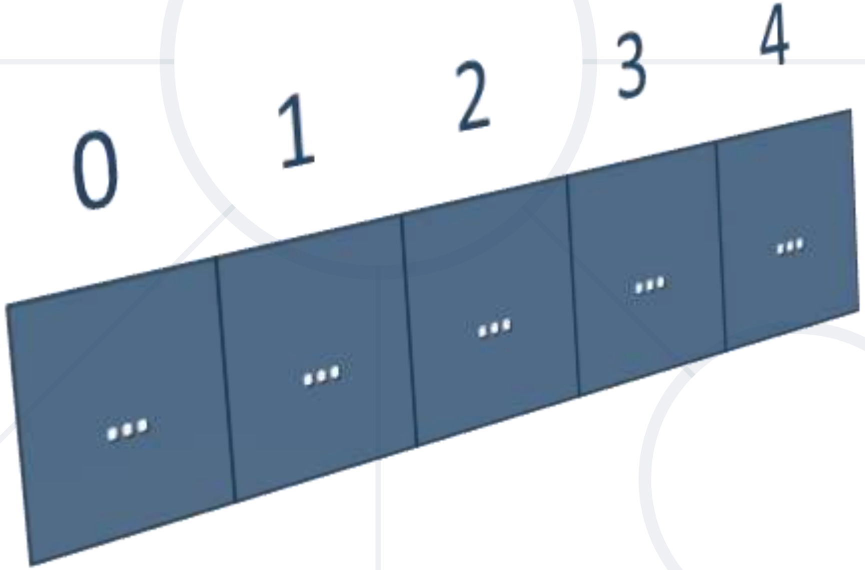
Arrays

Fixed-Size Sequences of Elements



SoftUni Team

Technical Trainers

Software University

http://softuni.bg

Table of Contents

SoftUni Foundation

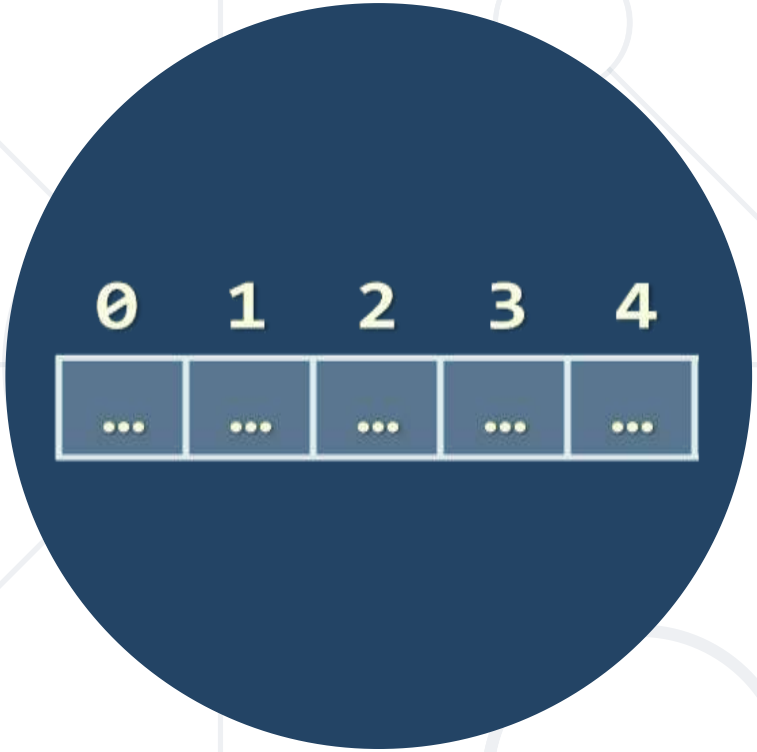
1. Definition
2. Operations
3. Array Iteration
4. For-in and For-of loops

Have a Question?

SoftUni Foundation

sli.do

#tech-fund

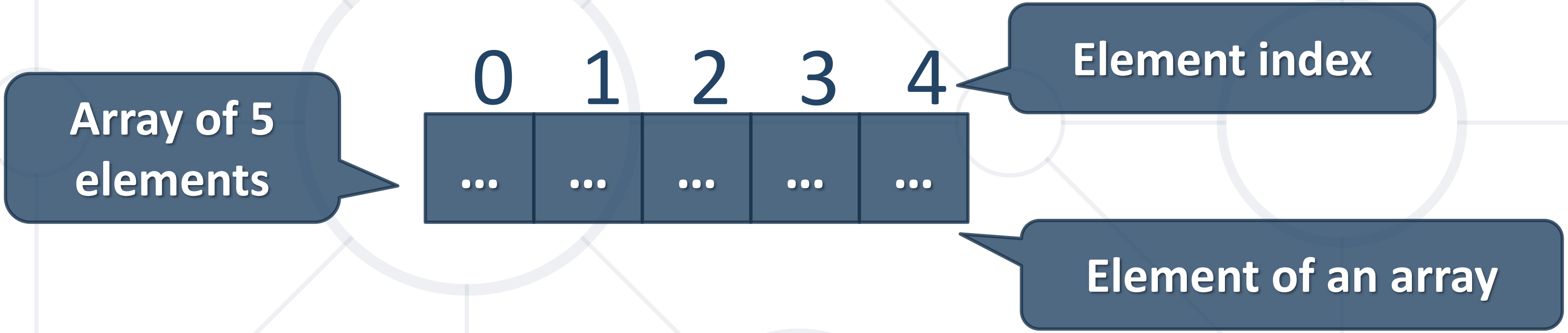


Arrays in JS

Definition and Simple Usage

What are Arrays?

- In programming, an **array** is a **sequence of elements**



- We can store **multiple values** in one variable
- Elements are numbered from **0** to **length-1**
- Arrays have **variable size** (**Array.length**) can be resized (unlike C# / Java)

Creating Arrays

- **Creating** an array of numbers:

```
let numbers = [1, 2, 3, 4, 5];
let names = [];
```

We are creating an Array using the literal **[ ]**

- **Accessing** array elements by index:

```
console.log(numbers[0]); // 1
```

The **[n]** operator accesses elements by **index**

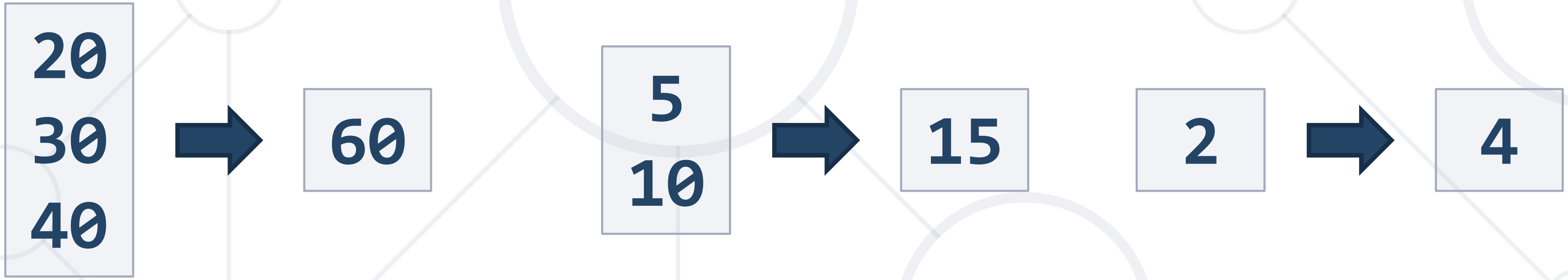
- **Assigning values** to the array elements:

```
numbers[3] = numbers[1] + numbers[2];
console.log(numbers.length); // 5
console.log(numbers); // [1, 2, 3, 5, 5]
```

The **length** holds the number of array elements

Problem: Sum First and Last Array Elements

- You are given array of strings holding numbers
  - Calculate and print the sum of the first and the last elements



```
function sumFirstAndLast(arr) {
  console.log(Number(arr[0]) + Number(arr[arr.length - 1]));
}
```

Check your solution here: https://judge.softuni.bg/Contests/1243/Arrays-Lab

Days of Week – Example

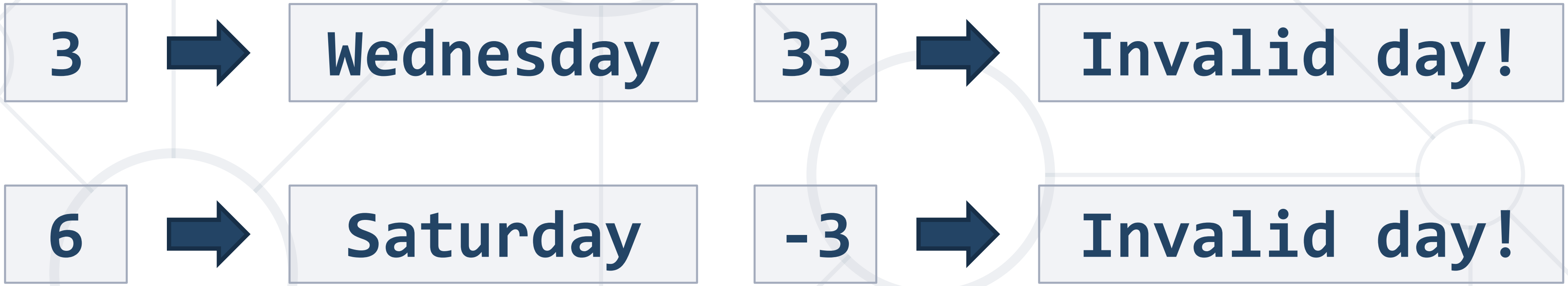
- The days of week can be stored in array of strings:

```
let days = [
  "Monday",
  "Tuesday",
  "Wednesday",
  "Thursday",
  "Friday",
  "Saturday",
  "Sunday"
];
```

Index	Value
days[0]	Monday
days[1]	Tuesday
days[2]	Wednesday
days[3]	Thursday
days[4]	Friday
days[5]	Saturday
days[6]	Sunday

Problem: Day of Week

- Write a program which receives a number and prints the corresponding name of the day of week (in English).
- If the number is not a valid day, print "Invalid day!"



```
3 → Wednesday
33 → Invalid day!
6 → Saturday
-3 → Invalid day!
```

Check your solution here: https://judge.softuni.bg/Contests/1243/Arrays-Lab

Solution: Day of Week

```
function dayOfWeek(day){
  let days = [ "Monday", "Tuesday", "Wednesday", "Thursday",
               "Friday", "Saturday", "Sunday" ];'
  if (day >= 1 && day <= 7)
    console.log(days[day - 1]);
  else
    console.log("Invalid day!");
}
```

The first day in our array is on index 0, not 1.

Check your solution here: https://judge.softuni.bg/Contests/1243/Arrays-Lab

Arrays of Different Types

```
// Array holding numbers
let numbers = [10, 20, 30, 40, 50];
```

```
// Array holding strings
let weekDays = ['Monday', 'Tuesday', 'Wednesday',
  'Thursday', 'Friday', 'Saturday', 'Sunday'];
```

```
// Array holding mixed data
let mixedArr =
  [20, new Date(), 'hello', {x:5, y:8}];
```

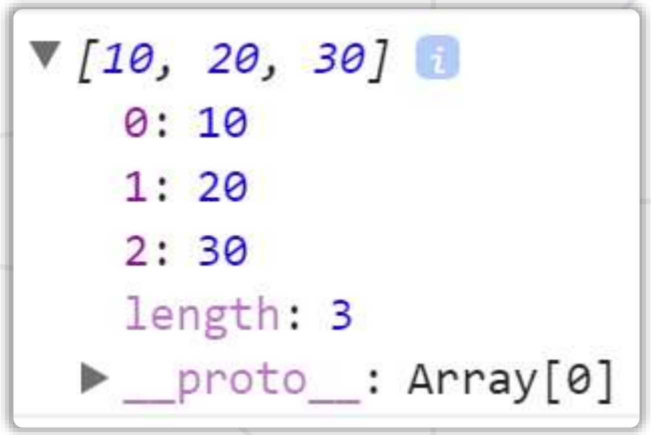
Operations

Basic array operations and printing

Simple Usage

- **Replace** an element value:

```
let arr = [10, 20, 30];
arr[0] = 5; // Elements can be modified
console.log(arr); // [5, 20, 30]
```



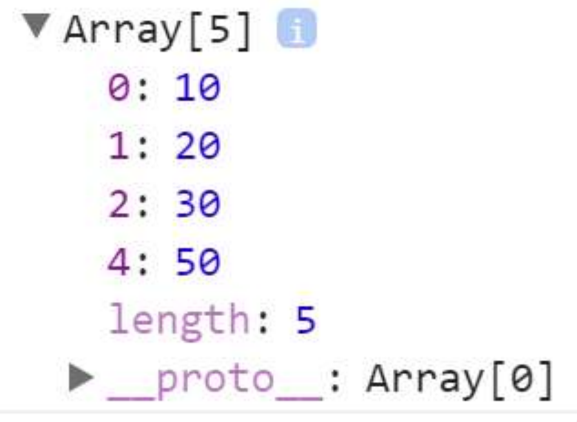
```
▼ [10, 20, 30]
    0: 10
    1: 20
    2: 30
    length: 3
  ▶ __proto__: Array[0]
```

- Check if the array **contains** the specified element:

```
console.log(arr.includes(20)); // true
console.log(arr.includes(0)); // false
```

JS Arrays and Invalid Positions

```
let nums = [10, 20, 30];
nums[4] = 50; // Will resize the array
console.log(nums); // [10, 20, 30, ,50]
console.log(nums.length); // 5
console.log(nums[3]); // undefined
```



```
▼Array[5]
   0: 10
   1: 20
   2: 30
   4: 50
   length: 5
  ▶__proto__: Array[0]
```

```
console.log(nums[-5]); // undefined
nums[-5] = -5; // Will not resize the array
console.log([nums[-5], nums.length]); // [-5, 5]
```

Pushing Elements in Array

- The right way to **add** elements in a JS array is to use **push**:

```
let arr = [10, 20, 30];
arr.push(40); // Adds an element at the end
console.log(arr); // [10, 20, 30, 40]
```

Array Iteration

Using a for Loop

Printing Arrays on the Console

- To print all array elements, a for-loop can be used

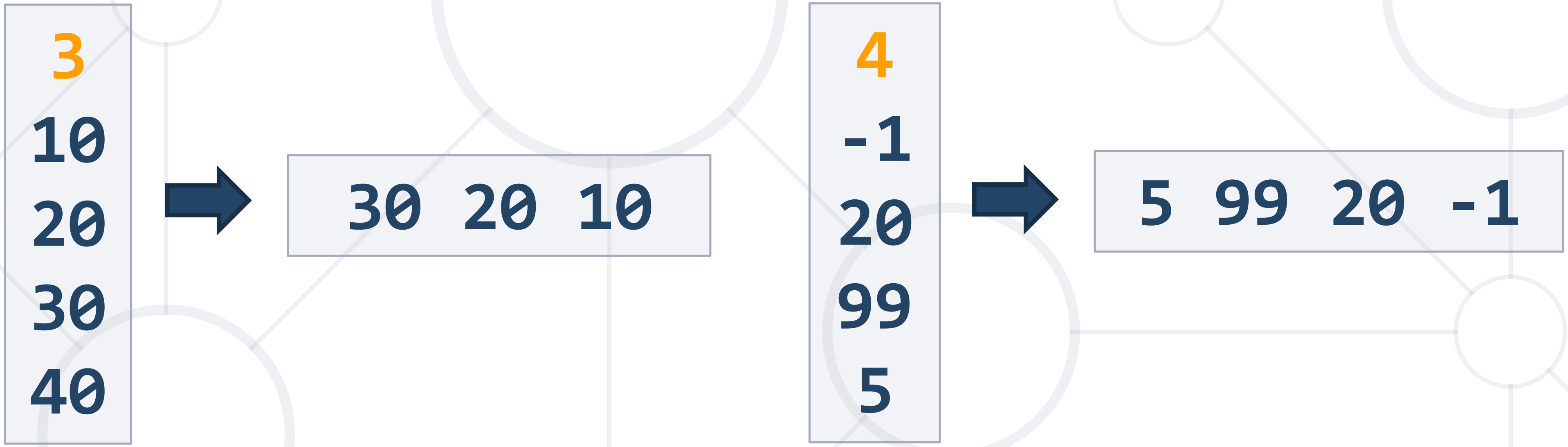
```
let capitals = ['Sofia', 'Washington', 'London'];

for (let i = 0; i < capitals.length; i++){
 console.log(capitals[i]);
}
```

- Print array elements using **toString()**

```
console.log(capitals.toString())
// Sofia, Washington, London
```

Problem: Reverse an Array of Numbers

- Receive a number **n** and an **array** of elements, **create** a **new** array with **n** numbers, **reverse** it and print its elements on a single line, space-separated:



```
3
10
20
30
40
```

```
30 20 10
```

```
4
-1
20
99
5
```

```
5 99 20 -1
```

Check your solution here: https://judge.softuni.bg/Contests/1243/Arrays-Lab

Solution: Reverse an Array of Integers

```
function reverse(n, inputArr) {
  let arr = [];
  for (let i = 0; i < n; i++)
    arr.push(inputArr[i]);
  let output = '';
  for (let i = arr.length - 1; i >= 0; i--)
    output += `${arr[i]} `;
  console.log(output);
}
```

Check your solution here: https://judge.softuni.bg/Contests/1243/Arrays-Lab

Printing Arrays with for / join

- Use for-loop:

```
let arr = [ 1, 2, 3, 4, 5 ];
for (let i = 0; i < arr.length; i++)
    console.log(arr[i]);
```

- Use `join(separator):`

```
let nums = [ 1, 2, 3 ];
console.log(nums.join(', ')); // 1, 2, 3
let words = [ "one", "two" ];
console.log(nums.join(' - ')); // one - two
```

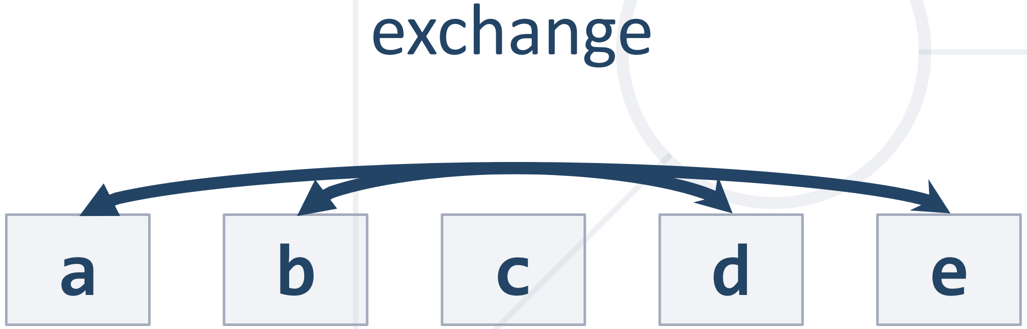
Problem: Reverse Array of Strings

- Receive an array of strings (space separated values), reverse it and print its elements:

`a b c d e` → `e d c b a`

`-1 hi ho w` → `w ho hi -1`

- Reversing array elements:



exchange

a b c d e

Check your solution here: https://judge.softuni.bg/Contests/1243/Arrays-Lab

Solution: Reverse Array of Strings

```
function reverse(elements) {
  for (let i = 0; i < elements.length / 2; i++)
    swapElements(elements, i, elements.length - 1 - i);
  console.log(elements.join(' '));
  function swapElements(arr, i, j) {
    let oldElement = arr[i];
    arr[i] = arr[j];
    arr[j] = oldElement;
  }
}
```

Check your solution here: https://judge.softuni.bg/Contests/1243/Arrays-Lab

For-in, for-of loops

Alternative way to iterate

For-of Loop

- Iterates through all **elements** in a collection
- Cannot access the current index

```
for (let el of collection) {
    // Process the value here
}
```

Print an array with For-of

```
let numbers = [ 1, 2, 3, 4, 5 ];
let output = '';
for (let number of numbers)
    output += `${number} `;
console.log(output);
```

```
1 2 3 4 5
```

For-in Loop

SoftUni Foundation

- Iterates through all **indexes** in a collection

```
let numbers = [ 5, 4, 3, 2, 1 ];
let output = '';
for (let index in numbers);
    output += `${index} `;
console.log(output);
```

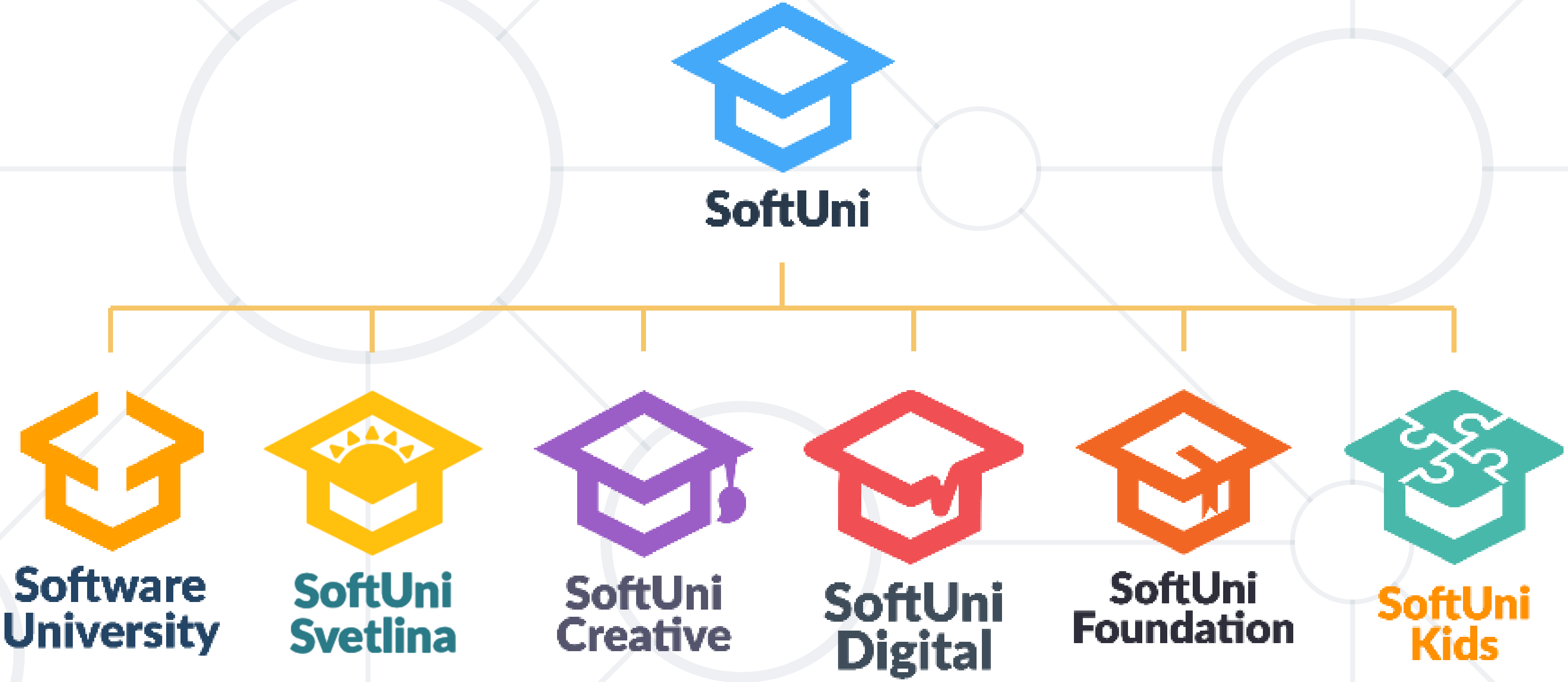
```
0 1 2 3 4
```

Live Exercises

Summary

- Arrays are **sequence** of elements
  - Elements are numbered from `0` to `length-1`
- Creating an array
- Accessing/Adding array elements
- Iterating through array elements
- For-of and For-in loops

Questions?



SoftUni

Software University

SoftUni Svetlina

SoftUni Creative

SoftUni Digital

SoftUni Foundation

SoftUni Kids

https://softuni.bg/courses/technology-fundamentals

SoftUni Diamond Partners

XSsoftware

SBTech
we know sports

telenor

SoftwareGroup
doing it right

NETPEAK

SmartIT

Postbank
Решения за твоето утре

SUPER HOSTING ®.BG

INDEAVR
Serving the high achievers

INFRAGISTICS®

LIEBHERR

æternity

codexio

SoftUni Organizational Partners

IS ИНФОРМАЦИОННО ОБСЛУЖВАНЕ

OneBit SOFTWARE

WORLD OF MYTHS

Lukanet.com

codexio

Trainings @ Software University (SoftUni)

- Software University – High-Quality Education and Employment Opportunities
  - softuni.bg
- Software University Foundation
  - http://softuni.foundation/
- Software University @ Facebook
  - facebook.com/SoftwareUniversity
- Software University Forums
  - forum.softuni.bg

License

- This course (slides, examples, demos, videos, homework, etc.) is licensed under the "Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International" license