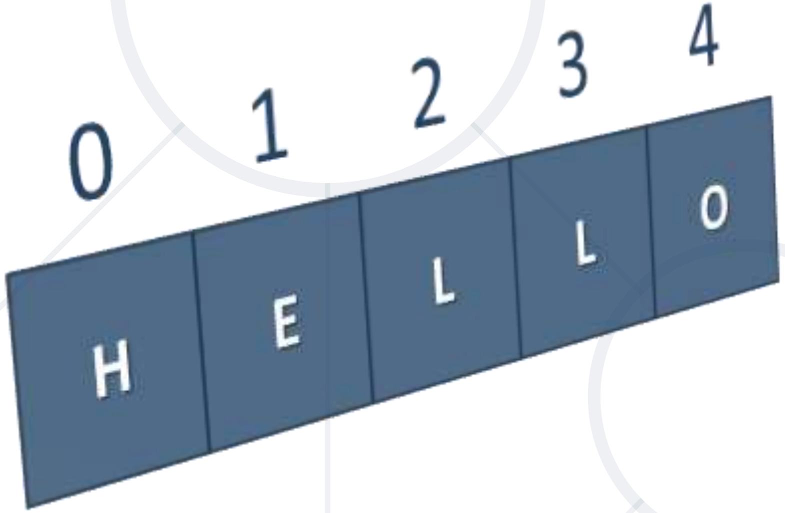
Strings and Text Processing



SoftUni Team

Technical Trainers

Software University

http://softuni.bg

Questions?

sli.do

#tech-fund

Table of Contents

SoftUni Foundation

1. What is a String?
2. Manipulating Strings
   - Substring
   - Replace
   - Delete
   - Split
3. Additional Functions

Strings

What is String?

What is String?

- Strings are sequences of characters (texts)
- Strings hold a sequence of characters
  - Like arrays, they have **length** (access by index)
- Strings are enclosed in three types of quotes:

```
let str = "Hello"; | let str = 'Hello'; | let str = `Hello`;
```

- Concatenated using the "**+**" operator:

```
let s = "Hello" + " " + "JS";
```

Strings are Immutable

- Strings are **immutable** (read-only) sequences of characters
- Accessible by index

```
let str = "Hello, JS";
let ch = str[2]; //l
ch = str.charAt(2); //l
```

Both declarations are the same

Problem: Print Characters

- Write a function that receives a string and prints all the characters on separate lines

Input	Output
AWord	A W o r d

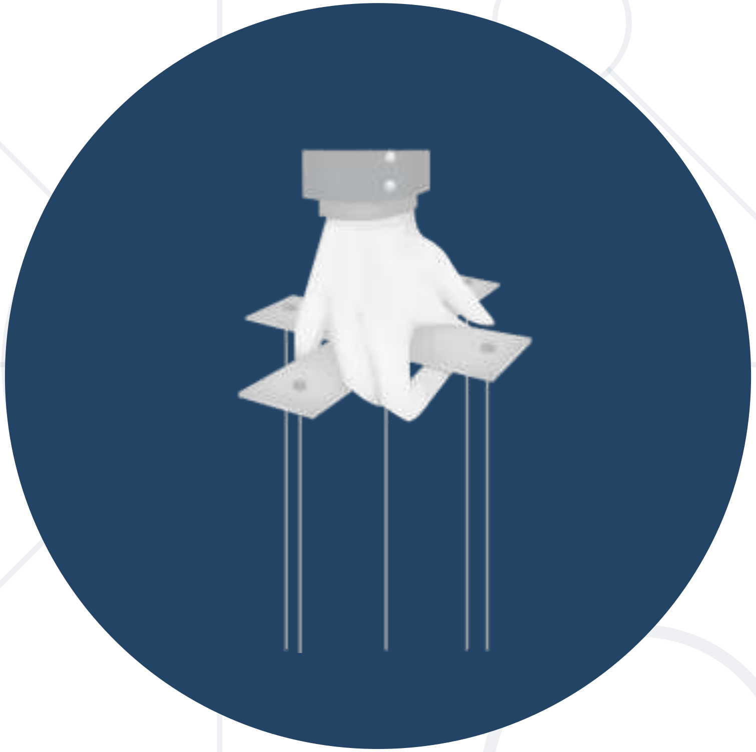
Solution: Print Characters

- Loop through the string and print each character

```
function solve(string) {
    for (let ch of string) {
        console.log(ch);
    }
}
```

- Test the function by calling it

```
solve("AWord");
```



Manipulating Strings

Concatenating

- Use the + or the += operators

```
let text = "Hello" + ", " + "JS!"; //"Hello, JS!"
```

```
let text = "Hello, ";
text += "John";                    //"Hello, John"
```

- Use the concat() method

```
let greet = "Hello, ";
let name = "John";
let result = greet.concat(name);
console.log(result); //"Hello, John"
```

Problem: Concatenation

- Write a function that receives an array of strings and prints a resulting string containing all of them

Input	Output
["First", "Second", "Third"	FirstSecondThird

Solution: Concatenation

- Loop through the array and add each element to a resulting string

```
function solve(arr) {
    let result = '';
    for (let word of arr) {
        //TODO: concatenate
    }
    console.log(result);
}
```

Searching for Substrings

- **Index of**:

```
let str = "I am JavaScript developer";
console.log(str.indexOf("Java")); // 5
console.log(str.indexOf("java")); // -1
```

- **Last index of:**

```
let str = "Intro to programming";
let last = str.lastIndexOf("o");
console.log(last); // 11
```

Extracting Substrings

- **Substr**:

```
let str = "I am JavaScript developer";
let sub = str.substr(5, 10);
console.log(sub); // JavaScript
```

Receives **start index** and **count**

- **Substring**:

```
let str = "I am JavaScript developer";
let sub = str.substring(5, 10);
console.log(sub); // JavaS
```

Receives **start index** and **end index** (exclusive)

Problem: Substring

- Write a function that receives a string and two numbers. The numbers will be a starting index and count of elements to substring

Input	Output
ASentance 1 8	Sentnace

Solution: Substring

- Create a new string that takes the needed amount of elements from the given string

```
function solve(string, startIndex, count) {
    let result = //TODO
    console.log(result)
}
```

String Operations (2)

- **Split**:

```
let text = "I love fruits";
let words = text.split(' ');
console.log(words); // ['I', 'love', 'fruits']
```

- **Includes**:

```
let text = "I love fruits.";
console.log(text.includes("fruits")); //True
console.log(text.includes("banana")); //False
```

String Operations (3)

- **Replace** – replaces first occurrence
  - The result is a new **string**

```
let text = "Hello, john@softuni.bg, you have been using
john@softuni.bg in your registration.";
    let replacedText = text.replace(".bg", ".com");
    console.log(replacedText);
//Output:
//Hello, john@softuni.com, you have been using
john@softuni.bg in your registration.
```

Problem: Censored Words

- Write a function that receives a text and a word. Find all occurrences of that word in the text and replace them with the corresponding amount of '*'

Input

`A small sentence with some words`
`small`

Output
`A ***** sentence with some words`

Solution: Censored Words

- Save the new text in a new variable

```
function solve(text, word) {
    let censored = text.replace(word, repeat(word));
    while (censored.includes(word)) {
        censored = censored
            .replace(word, repeat(word));
    }
    //TODO: create the repeat function
}
```

- The repeat function should take the length of the word and return that amount of stars '*'

Additional Functions

Checking, Padding, Removing Spaces

Starts with/Ends with

- Use **startsWith()** to determine whether a string **begins** with the characters of a specified substring

```
let text = "My name is John";
console.log(text.startsWith('My')); // true
```

- Use **endsWith()** to determine whether a string **ends** with the characters of a specified substring

```
let text = "My name is John";
console.log(text.endsWith('John')); // true
```

Padding at the Start and End

- Use **padStart()** to add to the current string **another substring** at the **start** until a **length** is reached

Receives **length** and **substring**

```
let bitAsStr = "010";
console.log(text.padStart(8, '0')); // 00000010
```

- Use **padEnd()** to add to the current string **another substring** at the **end** until a **length** is reached

```
let sentence = "He passed away";
console.log(text.padEnd(20, '.'));
// He passed away......
```

Trimming Strings

- Use **trim()** method to remove **whitespaces** (spaces, tabs, no-break space, etc.) from **both ends** of a string

```
let text = "   Annoying spaces       ";
console.log(text.trim()); // "Annoying spaces"
```

- Use **trimStart()** or **trimEnd()** to remove whitespaces **only** at the beginning or at the end

```
let text = "   Annoying spaces       ";
text = text.trimStart(); text = text.trimEnd();
console.log(text); // "Annoying spaces"
```

Repeating Strings

- Use **repeat()** to construct and return a **new string** which contains the **specified number** of copies of the string on which it was called

```
let n = 3;
for(let i = 1; i <= n; i++) {
  console.log('*'.repeat(i));
}
```

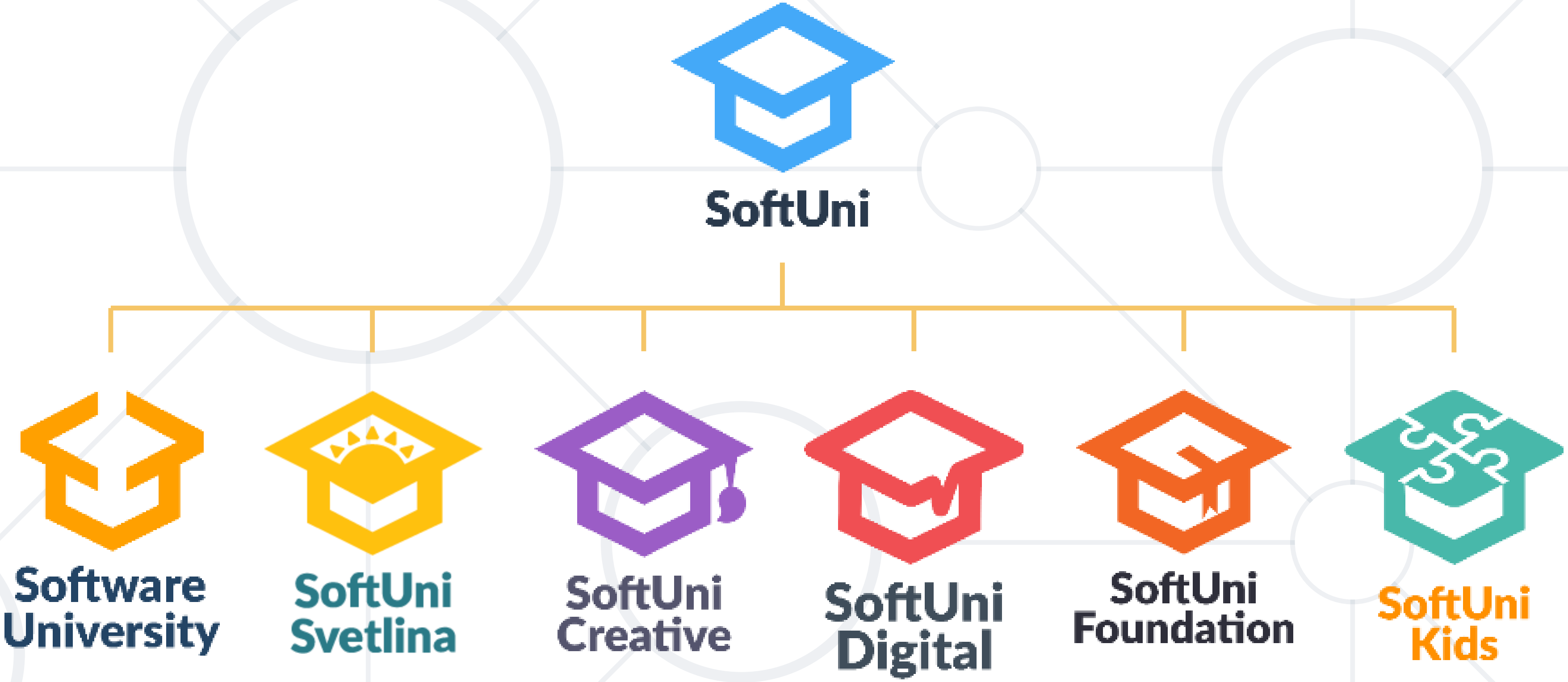
```
// *
// **
// ***
```

Live Exercises

Summary

- Strings are **immutable** sequences of Unicode characters
- String processing methods
  - concat()
  - indexOf()
  - includes()
  - substring()

Questions?


SoftUni

Software University

SoftUni Svetlina

SoftUni Creative

SoftUni Digital

SoftUni Foundation

SoftUni Kids

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