香港中文大學
The Chinese University of Hong Kong

CSCI2720 2023-24 Term 1: Building Web Applications

Lab 4: More on JavaScript

Dr Colin Tsang

Outline

- Using the console
  - Arrays
  - Arrow functions

- Event with DOM Tree

Using the console

- For consistency, we will use Google Chrome for our development use
- Start the console in Developer Tools
  - Windows: *ctrl* + *shift* + *J*
  - Mac: *command* + *option* + *J*

- To test JS code, you can run directly in this console, or output debug messages with **console.log(…)**

- More on: https://developer.chrome.com/docs/devtools/

Task 1: remove duplicated values in array

- Suppose an unsorted array with duplicated values:
  - [1, 2, 3, -21, 2, 1, 4]
- We want to remove the duplicated values. Now the new array becomes:
  - [1, 2, 3, -21, 4]
- Note that in the new array, the elements order is not important:
  - [-21, 1, 2, 3, 4] is also a valid output

Task 1: remove duplicated values in array

- Download *lab04.js* from BlackBoard. Fill in the TODO part:

```
function uniq(arr) {
    var ans;
    //TODO
    return ans;
    }
```

- There are many ways. Try to do it with:
  - **.sort()**
  - **.includes()**
  - **.filter()** and **.indexOf()**
- References: https://www.w3schools.com/jsref/jsref_filter.asp

Task 2: counting string length in an array

- Recall: functions

```
let func = function(arg1, arg2, ..., argN) { return expression; };
```

- An arrow function is a syntactic sugar:

```
let func = (arg1, arg2, ..., argN) => expression;
```

- Example: takes the sum of two numbers

```
let sum = (a, b) => a + b;
```

Task 2: counting string length in an array

- Now, given an array of strings, can we count the length of each string? (use arrow function and **.map()** and **.length**)
  - e.g., for the input array
    - `const materials = ['Hydrogen', 'Helium', 'Lithium', 'Beryllium’];`
  - The expected output is [8, 6, 7, 9]

- `console.log( materials.map(… => …));`
  - Try to fill in the “…”

Task 3 – build a magic image

- Download *lab04.html* from BlackBoard.
- In *lab04.html*, there is an image of magic book. Please add the following functionalities according to events:
  - On *mouse over*: showing a paragraph *"the mouse is not the secret!"*
  - On *mouse out*: the paragraph disappears.
  - On *key down*: if you are pressing *"s"*, show a paragraph *"s is the secret!"*; otherwise, show *"this key is not the secret"*.
  - On *key up*: the paragraph disappears.

Task 3 – build a magic image

- Try to implement with your knowledge of DOM and events.
- You can assume that the mouse event and key event will not happen at the same time.
- References:
  - https://developer.mozilla.org/en-US/docs/Web/API/KeyboardEvent/code

Submission

- No submission is needed for labs
- What you have done for this lab could be useful for your assignments
- Please keep your own file safely