

# WEEK09 – FORM & JAVASCRIPT

---

WEN-BIN JIAN

DEPARTMENT OF ELECTROPHYSICS, NATIONAL CHIAO TUNG UNIVERSITY

# OUTLINE

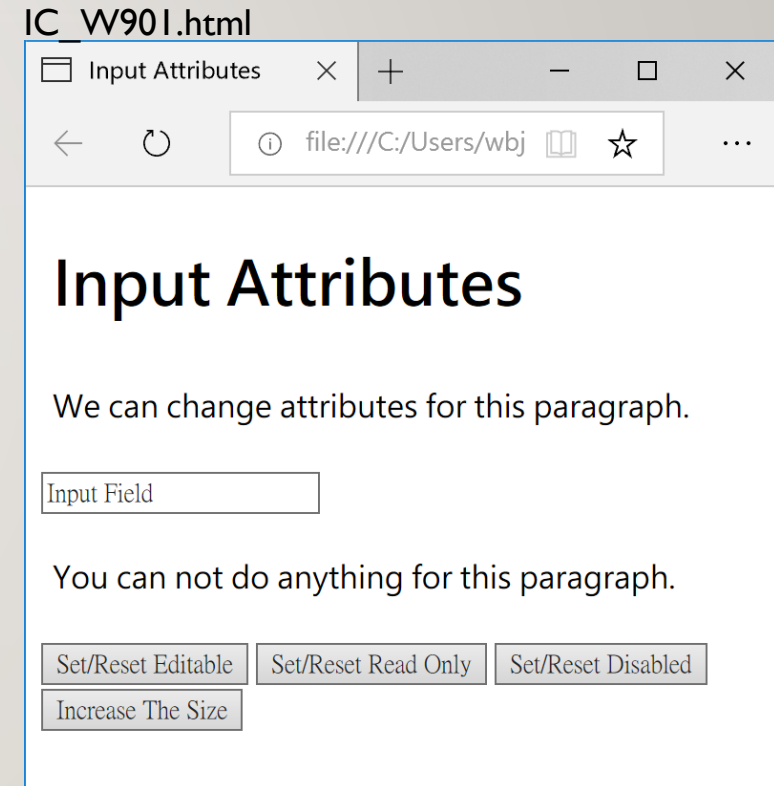
---

- 1. The HTML Input Attributes**
- 2. The HTML5 Input Attributes**
- 3. The Elements of Select/TextArea/Button**
- 4. The HTML5 Elements of DataList/Output/Progress**
5. Integrated Development Environment for JavaScript
6. Basic Concept & Reserved Words
7. Variables & The typeof Operator

# HTML INPUT ATTRIBUTES

---

- The attribute of **value** indicates the initial value for the field.
- The attribute of **readonly** indicates that the input field is not editable.
- The attribute of **disabled** indicates that the input field is not usable.
- The attribute of **size** indicates the size of the input field.
- The attribute of **maxlength** indicates the maximum input characters for the field.



# HTML INPUT ATTRIBUTES – HTML5

---

- The attribute of **autocomplete** specifies whether a input form/field is autocomplete on/off.
- The attribute of **autofocus** specifies that the field gets focus when the page loads.
- The attribute of **form** specifies the form that a input field belongs to.
- The attribute of **action** specifies the URL of a file for further processing.
- The attribute of **formenctype** specifies how the form data is encoded.

# HTML INPUT ATTRIBUTES – HTML5

---

- The attribute of **multiple** allows for entering more than one value in the <input> value.
- The attribute of **pattern** specifies a regular expression for the <input> value.
- The attribute of **placeholder** specifies a hint to the <input> value.
- The attribute of **required** specifies that the input field must be filled before submitting the form.
- The attribute of **step** specifies the legal number intervals for an <input> type of number.

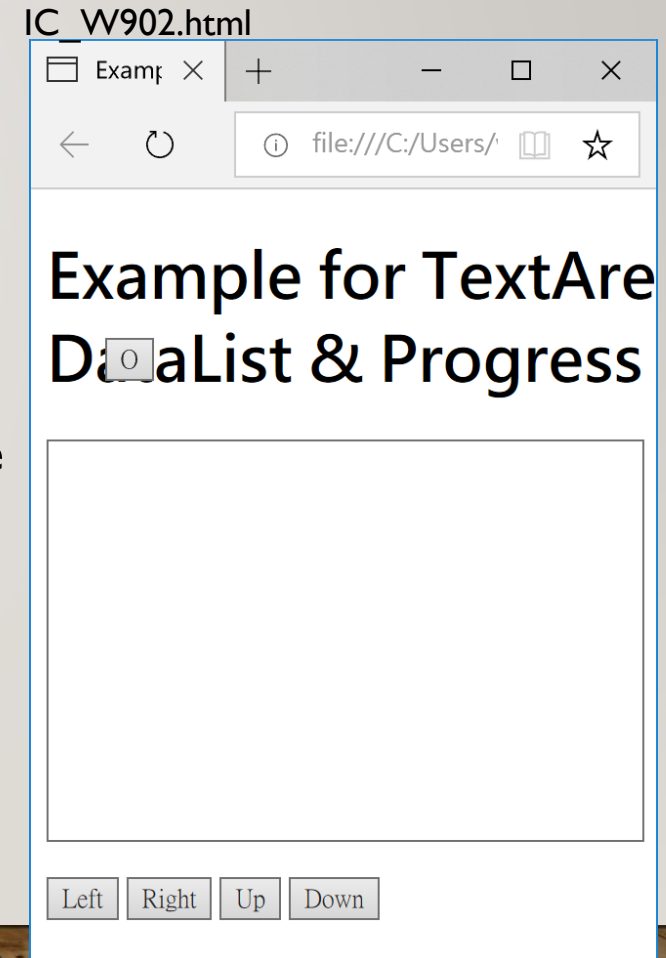
# HTML FORM ELEMENTS – TEXTAREA & BUTTONS

---

- The element of `<textarea>` specifies a multi-line input field.
- The attribute of `rows` specifies the visible number of lines.
- The attribute of `cols` specifies the visible width of the text area.
- The element of `<button>` defines a clickable button.
- The attribute of `onclick` defines the event processing codes when the button is clicked.

Useful attributes for other elements:

`position: fixed;`  
`left: ?; top: ?;`



# HTML INPUT ATTRIBUTES – HTML5

---

- The attribute of **formmethod** defines the HTTP method for sending data to the action URL.
- The attribute of **target** specifies where to display the response.
- For the input type of **image** (like a submit button), the attributes of **width** and **height** specify its dimension.
- The attribute of **list** specifies the id of a <datalist> object.
- The attributes of **min** and **max** specify the range for input types of date, number, ...

# HTML FORM ELEMENTS - SELECT

- The `<select>` element defines a drop-down list.
- The element in the list is defined in the region between `<option>` and `</option>`.
- The default selection is defined by an attribute of `selected` as `<option value="val" selected>`.
- The number of visible options is defined by the attribute of `size` in `<select size="n">`.
- The attribute of `multiple` for `<select>` is used for multiple selections.





# HTML5 FORM ELEMENTS – DATALIST, OUTPUT, PROGRESS

---

- The element of **datalist** is used following the input element.
  - `<input list="list_name">`
  - `<datalist id="list_name"> <option value="..."> <option value="..."> ... </datalist>`
- The element of **<output>** represents the result of a calculation. The calculation is expressed in the form with the attribute of **output**.
- The element of **<progress>** represents the progress of a task.

# OUTLINE

---

1. The HTML Input Attributes
2. The HTML5 Input Attributes
3. The Elements of Select/TextArea/Button
4. The HTML5 Elements of DataList/Output/Progress
- 5. Integrated Development Environment for JavaScript**
- 6. Basic Concept & Reserved Words**
- 7. Variables & The typeof Operator**

# A PROGRAMMING LANGUAGE

- JavaScript was developed since 1995.

---

- You can run it in html files on any kinds of browsers.
- You can run it in a standalone way using [Node.js](#) that can be used like PHP as a server side responding language.
- Let's start learning language in html files
  - Syntax, Statements, Reserved Words
  - Variables, Data type, Range of Values, Number Conversions, String, Object, Unary Operators, Bitwise Operators, Boolean Operators, Calculation, Equality Operators, Conditional Operators, Assignment Operators, Comma Operator
  - Flow-control, if, do-while, while, for, for-in, label, break & continue, with, switch
  - Functions, arguments, no overloading

# THE PROGRAMMING ENVIRONMENT

- Write codes in any text editor.
- Before, you got response from sequential DOS commands. You can get input and response to output in DOS console.
- Now you need to use HTML elements (graphical interfaces) to process input and output.
- Here we propose to use input fields for input and paragraph for output.
- Our integrated development environment (IDE) program consists of a browser and an editor.

IC\_W904.html

```
<body>
<p> How many lines of stars do you want to play?
</p>
<p> <input id="myinput" type="number" onchange="response();">
</p>
<p id="outscr">
</p>
<p>
<script>
    document.write("Hello World!");
    function response(){
        let loops = parseInt(document.getElementById("myinput").value);
        let output = document.getElementById("outscr");
        let outstring = "";
        for (let i=1; i<=loops; i++){
            for (let j=1; j<=i; j++){
                outstring += "*";
            }
            outstring += "<br>";
        }
        output.innerHTML = outstring;
    }
</script>
</p>
</body>
</html>
```

# BASIC CONCEPT OF THE LANGUAGE

---

- JavaScript is a case sensitive language. A variable 'mike' is different from 'Mike'. The function `typeof` cannot be mistyped as `typeOf`. The wrong usage will cause "no response" on your html page. Thus it is difficult to debug your codes.
- Comments in JavaScript are like those in C language. A line comment is started with `//` and block comments are quoted by `/* ... */`.
- A declaration statement: `var sum;` // Here no initial value is assigned to 'sum'. You may get trouble if you use the content of the 'sum'.
- You can assign a value in the next line: `sum = 0;`
- You can also declare a variable with an initial value: `var sum = 0;`

# BASIC CONCEPT OF THE LANGUAGE

---

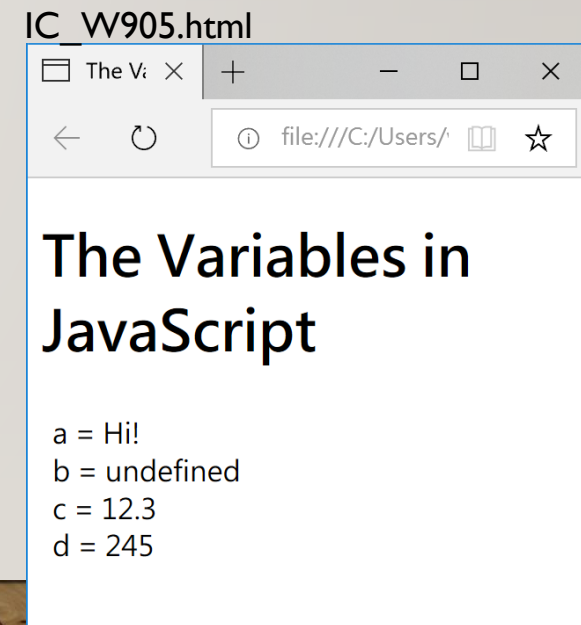
- A single-line conditional statement:
  - `if (x==1) window.alert("You got it");`
- Do several lines of commands in the conditional statement:
  - `if (x==1){`
  - `x=x+1;`
  - `window.alert("That's right.");`
  - `}`

# RESERVED WORDS

<b>break</b>	<b>do</b>	<b>instanceof</b>	<b>typeof</b>
case	else	new	var
catch	finally	return	void
continue	for	switch	while
debugger	function	this	with
default	if	throw	delete
in	try		
implements	package	public	interface
private	static	let	protected
yield			

# VARIABLES

- In JavaScript, variables are loosely typed. The single word 'var' is used to declare all kinds of variables. For a local variable, you may use 'let' to declare it.
- In JavaScript, the variables are objects. Before you initialize it, it is a null entity. You can give a value to assign the type of the variable.
  - `var a = "Hi!";` // here the variable a is a string (an array of characters)
  - `var b;` // a undefined object
  - `var c = 12.3;` // a float number
  - `var d = 245;` // an integer
  - `var a = 1, b = 2, c = 3;` // you can declare several variables with initialized values in one line
  - `function afunc(){ var a="12"; // local variable }`
  - `function afunc(){ a="12"; // global variable }`



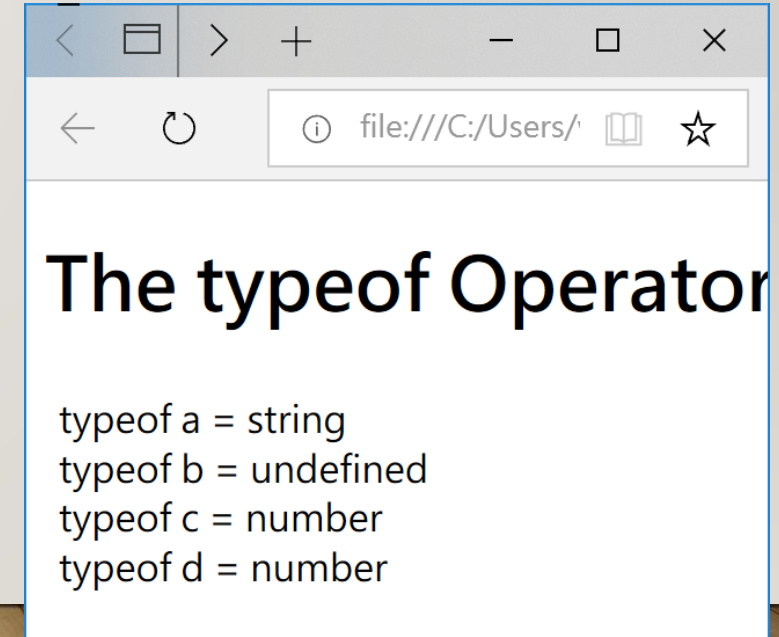


# THE TYPEOF OPERATOR

---

- The **typeof** operator is used to check the variable type.
- Before initializing the variable, the type is undefined.
- After you give an initial value, the variable may have each of the following types.
  - Null, Number, Boolean, String, ...
- Use the **typeof** operator to check and to learn variable types.

IC\_W906.html



```
typeof a = string
typeof b = undefined
typeof c = number
typeof d = number
```

# EXERCISE

---

1. Please design a form to collect users' data including name, gender, age, birthday, blood type, country, county/city, address. Try to use all kinds of input types for your form. Remember to give style setting for beautiful appearance. Give a button for user to see the complete input information by using `windows.alert`.
2. Please use two progress bars to control the position of one button on the html page. You need to get the attribute value of your progress bar.
3. Please prepare a html document with the header, navigation menu, body, and footer. Please put some contents in it. Please put some color inputs in the body so that the user can change the background, the border, and the text colors of the header, the menu, the body, or the footer.