HTML 5
File API

Bok, Jong Soon
javaexpert@nate.com
https://github.com/swacademy/HTML5

Using files from web applications

- Was added to the DOM in HTML5.
- Is possible for web content to ask the user to select local files.
- Is possible to read the contents of selected local user files.
- This selection can be done by either using :
  - An HTML `<input>` element
  - or
  - By drag and drop.

Accessing selected file(s)

- Makes it possible to access a **`FileList`** containing **`File`** objects.

```
<input type='file' id='file'>
```

- Accessing one selected file using a classical DOM selector:

```
var selectedFile =
    document.getElementById('file').files[0];
```

- Accessing selected file(s) on a **`change`** event :

```
<input type='file' id='file'
                onchange='handlerFiles(this.files)'>
```

Accessing selected file(s) (Cont.)

- If want to select multiple files, simply use the **`multiple`** attribute on the **`input`** element :

```
<input type="file" id="files" multiple
                onchange="handleFiles(this.files)">
```

- In this case, the file list passed to the **`handleFiles()`** function contains one **`File`** object for each file the user selected.

Accessing selected file(s) (Cont.)

- You need to use **`EventTarget.addEventListener()`** to add the **`change`** event listener, like this:

```
1  var inputElement = document.getElementById("file");
2  inputElement.addEventListener("change", handleFiles, false);
3  function handleFiles() {
4    var fileList = this.files; /* now you can work with the file list */
5  }
```

Getting information about selected file(s)

- The `FileList` object provided lists all of the files.
- Each specified as a `File` object.
- Can determine how many files the user selected by checking the value of the file list's `length` attribute :

```
var numFiles = files.length;
```

Getting information about selected file(s) (Cont.)

- Individual File objects can be retrieved by simply accessing the list as an array:

```
1  for (var i = 0, numFiles = files.length; i < numFiles; i++) {
2    var file = files[i];
3    /* code here */
4  }
```

File

- Provides information about files.
- Allows JavaScript in a web page to access their content.

Properties

- **`lastModified`**
  - Returns the last modified time of the file, in millisecond since the **`UNIX epoch`** (January 1st, 1970 at Midnight).
- **`lastModifiedDate`**
  - Returns the last modified **`Date`** of the file referenced by the **`File`** object.
- **`name`**
  - Returns the name of the file referenced by the **`File`** object.

Properties (Cont.)

- **`size`**
  - Returns the size of the file in bytes.
- **`type`**
  - Returns the **`MIME`** type of the file.
  - A string, containing the media type(MIME) indicating what type of the file is it for example "**`image/png`**" for PNG images.

FileReader

- lets web applications ***asynchronously***

  read the contents of files (or raw data buffers)

  stored on the user's computer

  using File or Blob objects

  to specify the file or data to read.

Constructor

- `FileReader()`
  - Returns a newly constructed `FileReader`.

Properties

- `error`
  - Represents the error that occurred while reading the file.
- `readyState`
  - A number indicating the state of the `FileReader`.
  - This is one of the following:

EMPTY	0	No data has been loaded yet.
LOADING	1	Data is currently being loaded.
DONE	2	The entire read request has been completed.

Properties (Cont.)

- `result`
  - The file's contents.
  - Is only valid after the read operation is complete
  - The format of the data depends on which of the methods was used to initiate the read operation.

Event handlers

- **`onabort`**
  - A handler for the **`abort`** event.
  - This event is triggered each time the reading operation is aborted.
- **`onerror`**
  - A handler for the **`error`** event.
  - This event is triggered each time the reading operation encounter an error.

Event handlers (Cont.)

- `onload`
  - A handler for the `load` event.
  - This event is triggered each time the reading operation is successfully completed.
- `onloadstart`
  - A handler for the `loadstart` event.
  - This event is triggered each time the reading is starting.

Event handlers (Cont.)

- `onloadend`
  - A handler for the `loadend` event.
  - This event is triggered each time the reading operation is completed (either in success or failure).
- `onprogress`
  - A handler for the `progress` event.
  - This event is triggered while reading a Blob content.

Methods

- `abort()`
  - Aborts the read operation.
  - Upon return, the `readyState` will be `DONE`.
- `readAsArrayBuffer()`
  - Starts reading the contents of the specified `Blob`, once finished, the `result` attribute contains an `ArrayBuffer` representing the file's data.
- `readAsBinaryString()`
  - Starts reading the contents of the specified `Blob`, once finished, the `result` attribute contains the raw binary data from the file as a string.

Methods (Cont.)

- **`readAsDataURL()`**
  - Starts reading the contents of the specified **`Blob`**, once finished, the **`result`** attribute contains a data:
  - URL representing the file's data.
- **`readAsText()`**
  - Starts reading the contents of the specified **`Blob`**, once finished, the **`result`** attribute contains the contents of the file as a text string.

Lab1 : File API

- Web Browsers
  - Edge, Firefox, Google Chrome, Opera, Safari
- Text Editors
  - Visual Studio Code, Notepad++, Editplus, etc…
- Files
  - fileapidemo.html

Lab1 : fileapidemo.html

```
<script>
  window.addEventListener('load', setup, false);
  function setup(){
    var myfiles = document.querySelector('#myfiles');
    myfiles.addEventListener('change', myChange, false);
  }
  function myChange(){
    var file = this.files[0];
    var str = '<ul>';
    for(var key in file){
      str += '<li>' + key + ' : ' + file[key] + '</li>';
    }
    str += '</ul>';
    document.getElementById('fileinfo').innerHTML = str;
  }
</script>
```

Lab1 : fileapidemo.html

```
25  <body>
26    <div>
27      File : <input type='file' id='myfiles' />
28    </div>
29    <h1>File Info</h1>
30    <div id='fileinfo'></div>
31  </body>
```

Lab1 : Result


Jong Soon
File API Demo
localhost:8080/fileapidemo.html
File :
Choose File
check.png
File Info
name : check.png
lastModified : 1521639836314
lastModifiedDate : Wed Mar 21 2018 22:43:56 GMT+0900 (Japan Standard Time)
webkitRelativePath :
size : 8831
type : image/png
slice : function slice() { [native code] }