MIS3690 - Web Technologies

Introduction to JavaScript

What is JavaScript?

- A programming language
  - originally used only in web browsers (with JavaScript engine)
  - now embeded in servers, usually via Node.js
- Client-side JavaScript
  - the main focus of this course
  - enabling interactive web pages
    - create dynamically updating content
    - control multimedia
    - animate images
    - and many other things
- Server-side JavaScript

How to add JavaScript

- Internal JavaScript
  - can be added in `<head>` or bottom of `body` (preferably)
  - we will be using this in class
- External JavaScript
  - create `.js` file
  - use `defer` / `async`
  - syntax:

```
<script src="script.js" defer></script>
```

- Inline JavaScript handlers
  - easy to connect event with element
  - it is bad practice to *pollute* your HTML with JavaScript (**not allowed!**)

JavaScript Examples

- Download *lec12-js-demo.html* from GitHub (*mis3690/resources/templates*)
- Open the file in web browser and interact
- Read the source code
  - any questions?

JavaScript - Basic Concepts

DOM - Document Object Model

- What is DOM?
  - the data representation of the objects that comprise the structure and content of a document on the web
  - a programming interface for HTML documents
- See example

Fundamental data types

- Document
  - the root
- Node
  - every object located within a document is a node of some kind
  - could be an *element* node, or an *attribute* node
- ***Element***
- NodeList
  - an array of nodes

What is an *event*?

When a page load happens, do play the video of a cat sliding into cardboard.

When a click happens, do submit my online purchase.

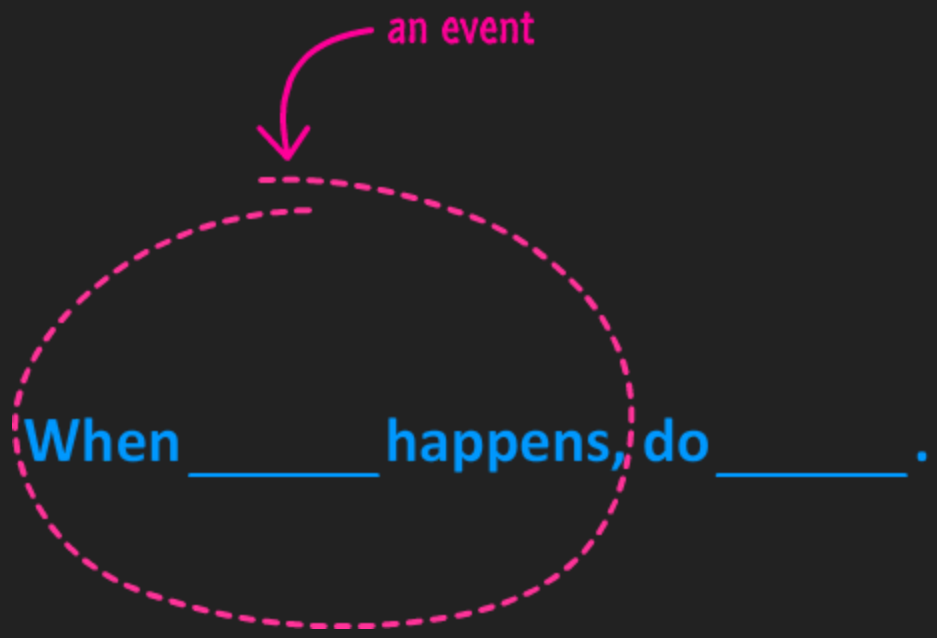
When a mouse release happens, do hurl the giant/not-so-happy bird.

When a delete key press happens, do send this file to the Recycle Bin.

When a touch gesture happens, do apply this old timey filter to this photo.

When a file download happens, do update the progress bar.

What is an *event*?



Handling events

- Event listener
  - `EventTarget.addEventListener()`
- Event handler
  - **on*event***

Registering on*event* handlers

- The on*event* handlers are properties on certain DOM elements to manage how that element reacts to events
- Two ways
  - adding an HTML *attribute* named on<eventtype>:

```
<button onclick="handleClick()">
<!-- again, it is bad practice -->
```

  - setting the corresponding property from JavaScript:

```
document.querySelector("button").onclick = function(event) {
  ...
}
```

Exercise

Modify *lec12-js-demo.html*, so it uses **the second way** to handle the event of clicking on the button

Object properties

- JavaScript treats everything as an ***object***
- *Properties* describe the characteristics of an object
  - use *dot notation*: `object.property`
    - examples:
      - `document.title` - the title property of a web page doc
      - `image.src` - the source property of the image element
  - different types of objects have different properties

Object methods

- *Methods* are functions that are performed by an object
  - think of them as verbs
- use *dot notation*: `object.method(arguments)`
  - `arguments` could be empty
- examples:
  - `document.getElementById("a")`
    - `document` - the object
    - `getElementById("a")` - the function (or method) that is part of this object
    - it gets the "puppet strings" to the element whose id is `"a"`
  - `console.log(message)`
    - outputs `message` to the web console
    - is an important way to debug

Exercise

Play with *lec12-js-demo.html* with what you just learned.

Questions?