COMPSCI 326 - Web Programming JavaScript DOM events

join on the Slack #q-and-a channel as well as Zoom
remember, you can ask questions of your teammates on your group Slack!
please **turn on your webcam** if you can
mute at all times when you aren't asking a question
([https://docs.google.com/document/d/1PROSgwoJqY1M8xV3r6qU6ESD2ERdwkHX4uj
ZnkO2JMM/edit?usp=sharing](https://docs.google.com/document/d/1PROSgwoJqY1M8xV3r6qU6ESD2ERdwkHX4ujZnkO2JMM/edit?usp=sharing))

Background resources:

videos
[35: What Are Event Listeners In JavaScript | JavaScript Events | JavaScript Tutorial | mmtuts](#)

web sites
[Events and the DOM - Web APIs | MDN](#)
[<script>: The Script element - HTML: Hypertext Markup Language](#)

Today: JavaScript DOM Events

First: review last class
Exercise today: interactive encoder/decoder (demo)

Encoder/decoder

Key: [cdefghijklmnopqrstuvwxy]

Text to encode: [hello world]   Encoded = [jgnnq yqtnf]
Text to decode: [   ]   Decoded = [   ]

[Go]

Events
- Last time, we had exactly one event: onClick(), when the button was pushed
- All DOM elements can have events and associated functions
- To trigger a function on an event, you add an *event listener*
  - `document.getElementById('key').`**`addEventListener`**`('eventname', yourFunction)`
  - [JavaScript DOM EventListener](#)
- NOTE! Can't do this at the top of your HTML in just a <script>
  - DOM still under construction
  - Browser executes scripts *concurrently* with rendering the page
  - Nice explanation here: https://blog.logrocket.com/how-browser-rendering-works-behind-the-scenes-6782b0e8fb10/
- Two options:
  - <SCRIPT> at the bottom of your HTML (just before </BODY>)
  - <SCRIPT **DEFER** SRC=""></SCRIPT> at the top (doesn't work for inline JavaScript) ← **strongly preferred, this is what we will do for realz**
    - It is bad practice to mix HTML and JavaScript; we're just doing this here for convenience in class. In your assignments, you will always use separate files

HTML

```html
<!DOCTYPE html>
<html lang="en">
 <head>
  <title>My Web Page</title>
  <!-- this is a comment -->
  <!-- scripts go here -->
 </head>
 <body>
  <!-- your document goes here -->
  <h1>Hello, World!</h1>
  <script>ADD EVENT LISTENERS HERE</script>
 </body>
</html>
```

Many different kinds of events (irritatingly - in HTML, these are all prefixed with "on" -- onClick):
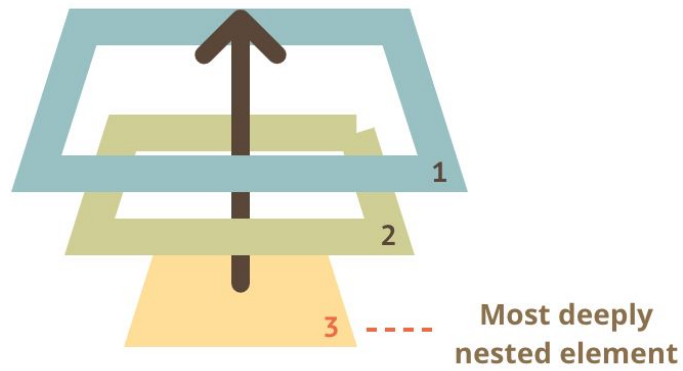- click → duh
- change → also duh
- focus → arrow/tab has moved inside
- blur → arrow/tab has moved out
- input → whenever user input is received
- **keyboard:** keydown (any key), keypress (not ALT, CTRL), keyup (end)
- **mouse:** click, dblclick, mousedown, mousemove, mouseover
- **drag** events, **clipboard** events, **media** events like video...

https://www.w3schools.com/tags/ref_eventattributes.asp

Used for interactivity, validation, etc.

- Can have multiple event listeners on the same object (this can be tricky)
- Event "bubble up" → start with the nearest object (like an input) and go to the containing object, etc. etc.
  When an event happens on an element, it first runs the handlers on it, then on its parent, then all the way up on other ancestors.

- You can explicitly stop propagation via `event.stopPropagation()`

Exercise!

https://docs.google.com/document/d/1kwrdoG_r7FLWIhzo4qlBUAHFZkT8ZeTdjaYKnK3FA-s/edit?usp=sharing