Web Programming

In-class activities
Week 4 – Manipulating the DOM with JavaScript

Prof. Josué Obregón

Department of Industrial Engineering- ITM

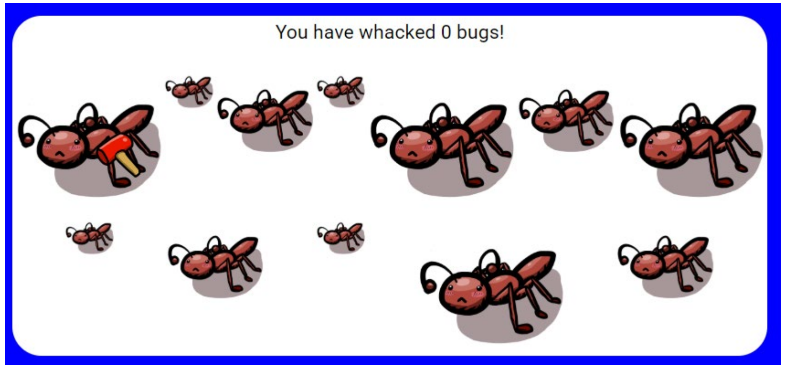
Seoul National University of Science and Technology

In-class activities

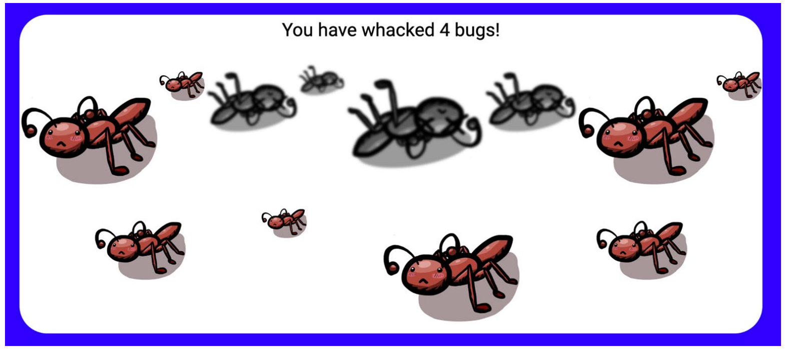
- 11:00 – 11:10 written quiz
- 11:10 – 11:50 Q&A and Hands-on exercise 1(with pair activity)
- 11:50 – 12:00 Break time
- 12:00 – 12:50 Hands-on exercise II

Exercise 1: Whack-a-Bug! (part 1)

- In this exercise, you will be implementing the whack-a-bug interactivity, which is triggered whenever a bug is clicked.
  - Changes the image of the bug that was clicked from `bug.png` to `bug-whacked.png` to signify it was whacked
  - Increment the whack count by 1



You have whacked 0 bugs!



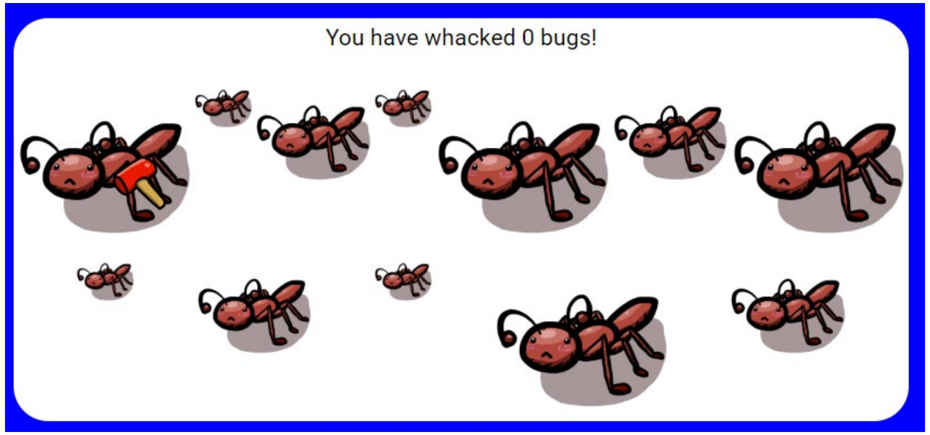
You have whacked 4 bugs!

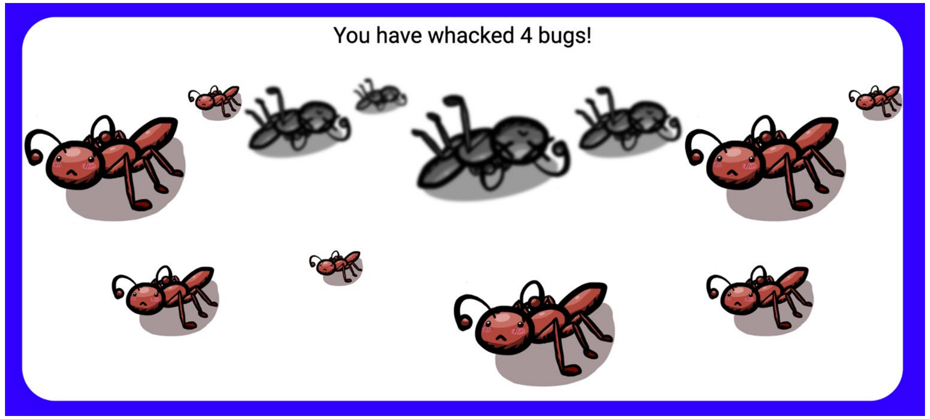
whack-a-bug.html

```
<section id="game">
   <p>You have whacked <span id="score">0</span> bugs!</p>
     <div id="bug-container">
        <img src="bug.png" alt="An ant-like bug">
        <img src="bug.png" alt="An ant-like bug">
                              …
        <img src="bug.png" alt="An ant-like bug">
     </div>
</section>
<footer>
  <p>Page and assets &copy; Sven Hansen</p>
</footer>
```

Exercise 1: Whack-a-Bug! (part 11)

- Download the starter code from e-class.
- You will implement the functionality to do the following:
  - o Changes the image of the bug that was clicked from `bug.png` to `bug-whacked.png` to signify it was whacked
  - o Add the whacked class to the image that was clicked
  - o Increment the whack count by 1 - each bug should only be counted once even if clicked multiple times
  - o If all the bugs have been whacked, change the text in the `#game p` tag to say "all bugs have been whacked"


You have whacked 0 bugs!


You have whacked 4 bugs!

Manipulating the classList of an Element

el.classList.add("class")	Adds the given class if not already present on the element.
el.classList.remove("class")	Removes the given class if it is present on the element.
el.classList.contains("class")	Returns whether the element has the given class.
el.classList.toggle("class")	Toggles the class on or off.

Exercise 2: Interactive Blog

- The HTML and CSS for a simple blog has been provided for you (download from e-class).
- You are tasked with writing adding interactivity to this page as follows:
  - When the button is clicked, an article should be appended to the #posts container. Inside should be a third level heading followed by a paragraph. The article should have the class .post added to it.
  - The third level heading text content should be the text "Date: " followed by the date submitted. The paragraph's text content should be the text "Entry: " followed by the entry submitted (It is not necessary to validate the specific format of the content)
    - Hint: to grab the text from a form element use .value
  - If a user double clicks on any of the blog posts, the post that was clicked should be removed from the page
  - The content in #date and #entry should be cleared
  - If the number of blog posts ever hits 3, the “add entry!” button should be disabled
    - Hint: use the querySelectorAll function to find the number of blog posts on the page