

Spring 2020 AIS TD Virtual Workshop 3 Guideline: **Website Accessibility**

Before we start

- Make sure you **download VS Code**
- The installation instructions for TD WS 3 [here](#).

Video for this Virtual Workshop

- Please go to our website to check it out. <https://www.temple-ais.org/>

General Concepts

- The fact is that millions of internet users have special needs, disabilities, and impairments that can make it difficult or even impossible for them to use certain types of websites.
- Hence, the main concept of Web Accessibility is to provide the same experience for everyone despite the conditions they are experiencing.
- Some common categories of impairments include:
 - **Visual Impairment:** This includes a partial or total inability to see or to perceive color contrasts.
 - **Hearing Impairment:** Some users have a reduced ability to hear.
 - **Motor Skills/Physical Disabilities:** Users may have difficulty moving parts of their bodies, including making precise movements (such as when using a mouse).
 - **Photosensitive Seizures:** Conditions such as epilepsy can cause seizures that are often triggered by flashing lights.
 - **Cognitive Disabilities:** There are also many conditions that affect cognitive ability, such as dementia and dyslexia.

Setting Up

Let's go ahead and get you all set up for this workshop! Yus!

Download and Unzip the **WS3 Code Folder**

- In the Drive Folder online, right click on the folder TD Workshop 3 → Select Download
- The folder will be downloaded to your device as a zip file.
- **Double click** on the zip file, or right click and select unzip to open the folder.
- Once you are done, we recommend you to **move the folder to your desktop screen** so it's easier for you to keep track of our next steps.

Connect the **WS3 Code Folder** to VS Code software

- Open your VS Code
- Go to your navigation menu bar on the top.
- Select **File** → Select **Open...** → Go to the directory that has your **WS3 Code Folder** → Select the Folder and click **Open**
- The folder should be opened on your VS Code Screen

Activate your Screen Reader

Screen readers are software programs that allow blind or visually impaired users to read the text that is displayed on the computer screen with a speech synthesizer.

***Notes:** You don't need to activate screen reader to work on these following exercises. We are just showing you how a screen reader works on your laptop! (So,... you can skip this if ya want!)

For Mac Users

- VoiceOver is a screen reader software built into Mac so you can use it for free.
- There are a few ways to turn on VoiceOver
 - **1) Using Siri**
 - Simply ask your Siri "Hey Siri, can you turn on VoiceOver?"
 - Siri will answer "Okay, I turned on VoiceOver"
 - **2) Using Shortcut**
 - Depends on your Mac versions, you can turn on VoiceOver by hitting either:
 - **Command-F5**
 - or **Fn-Command-F5**
 - If you encounter any issue, check this [Mac accessibility shortcuts](#) for more information!

- Once you open VoiceOver, you'll have the option to go through a **VoiceOver Quick Start** to learn about how you can navigate using only keyboard and without a mouse!
- If you want to start right away, just click the **V** key.

For Windows Users

- Narrator is a screen-reading app that's built into Windows 10, so there's nothing you need to download or install.
- NVDA and JAWS are the two most common screen readers on the market, but for demonstration purposes we'll use Narrator.
- **Ctrl + Window Key + Enter** to quickly turn on Narrator.

And you are all set and ready for our exercises yay :)

Exercise 1: Color Contrast

- Making sure text stands out against the background to help users with low vision
- Tools to check color contrast: Web AIM, Contrast Checker
- Alternatively, you can right click on a web page → Inspect → press **Ctrl + Shift + C (Mac: Command + Shift + C)** on a text element on the page
- As a rule of thumb, the visual presentation of text and images of text should have a contrast ratio of at least 4.5:1. If the text is bolded, or has a much bigger font (24px and above), the minimum contrast ratio is 3:1.
- Let's check the color contrast of the logo **"Cozee"**
 - You will notice that the color contrast is **2.06:1**
- Let's change the color of the logo to improve the color contrast
 - In **style.css**, take a look at **line 101**
 - The color is currently set to **#d9ae6a**. Let's change it to **#8A4500** for a darker shade of brown
 - Repeat the steps for the logo in the checkout page. To do this, you can navigate to **line 27** of **checkout.css**
 - Now, check the color contrast and notice how the ratio is much higher!
- For this exercise, change the color of **"AIS Technical Development Workshop 3"** text in the footer to increase the color contrast. You can do this by changing the color on **line 428** of **style.css**. Make sure to check the color contrast ratio after changing the color.
- **Submission:**
 - After you fix the code, **copy the entire code from line 428-434** and paste it in the submission form. Make sure to indicate the new color code (eg. #8A4500), and **take a screenshot** of the color contrast ratio!

Exercise 2: Link

- Making hypertext links accessible is one of the most basic and most important aspects of web accessibility
- In most browsers, the **Tab** key allows users to jump from link to link, and the **Enter** key allows users to select a link.

Exercise 2.1: Non-empty href attribute

- Links must have a **non-empty href** attribute and should be underlined.
- in **homepage.html**, take a look at **line 46**
- Even though we have the anchor tag `<a>`, we haven't specified the anchor tag to our **#about** section (line 95)
- Thus, the screen reader can't read this anchor tag → It will skip it completely
- For this exercise, let's add the href **#about** attribute to this form
- **Submission:** After you fix the code, **copy the entire code from line 46** and paste it in the submission form

Exercise 2.2: Underline link

- Links are easy to find because users understand that underlined text means that it's a link. The underlining draws their attention.
- In **homepage.html**, **line 106**, let's use the `<u>` tag to underline the link that leads to the external CDC website under Covid-19 section.
- **Submission:** After you fix the code, **copy the entire code from line 106** and paste it in the submission form .
- Now we will show you how Screen Reader will read the link with **Tab** and **Enter** keys

Exercise 3: The logical-semantic difference

- **** and **** tags are elements of HTML code that you can use to highlight important words.
 - **Bold ** makes text only visually look bold
 - **Strong ** semantically emphasizes the respective text
- Similarly, **<i>** stands for italic while **** stands for emphasis.
 - While both are typically (but not always) styled to display as italic
 - **** is used when you want to put more stress on a word or phrase.
- Let's change the tags so the screen reader can read the text in a "logical-semantic" way
- In **homepage.html**: line 101
 - Change ** ** tag to ** **
 - Change **<i> </i>** tag to ** **
- **Submission:** Copy the finished code in **line 101** and put it in the submission form!

Exercise 4: Media and Alt Text

- **Image:** you need **alt text** as replacement of image for non-visual users and/or if an image fails to load.
 - **Alt Text** announces the description of the image to users with screen reader
 - It is an **attribute** of the **** tag in HTML
- **Video:** you need to provide **transcripts** and **close captioning** to allow users who cannot hear to understand what is being said in a video.
 - You also need **audio description** to allow users who cannot see to visualize what is going on on the screen
- Let's add an alt text to one of the image elements!
 - In **product.html**, navigate to **line 110**
 - Inside the **** tag, add **alt= "Dalgona Quarantine Coffee"**
- To check, let's pretend we are missing the picture for **Dalgona Coffee** on **product.html**
 - Manipulate the **src** of the **img** so it won't show the right image file.
 - Refresh and look at the browser again → the image is now missing
- However, if you activate the Screen Reader → it will read out the alt text for this picture even though the image fails to load.
 - Turn on your Screen Reader
 - For Mac User → **Control + Option + [Right Arrow Key]** to move around the component

- For Windows User → **[Down Arrow Key]** to move around the component
- Add an **alt text** to **two** other images on **product.html**. You should also test it with your screen reader!
- **Submission:** for each image with an alt text that you just added:
 - Indicate the name of the image (ie. the product name)
 - Copy the respective line of code where you add the alt text
 - Indicate the line number of the code

Exercise 5: Table

- Data tables are used to organize data with a logical relationship in grids. Accessible tables need HTML markup that indicates header cells and data cells and defines their relationship.
- Assistive technologies use this information to provide context to users.
- In this exercise, we will make sure we provide the right tag, either `<th>` or `<td>` for the right data cell
- Let's go to **product.html**
 - Open it on browser
 - You will notice we have two tables
 - Look into the code
 - For Product Prices table - **line 241 and line 242**
 - For Shipping Fee - **line 262 and line 263**
 - This is not the correct tag since this should be **Table Header**
 - Let's fix both tables by replacing `<td>` with `<th>` for the first row.
- **Submission:** Copy the entire code for both tables and put it in submission form.

Exercise 6: Keyboard Interaction

- Put simply: **for a website to be accessible, it must work without the use of a mouse.**
- Users with mobility impairment and visual impairment use the **Tab key** to navigate the page
- Using the Tab key will ideally generate 'focus', which indicates the **interactive** element that the user is currently on.
- Let's take a look at the navigation menu
- Now let's then compare that to the checkout page: **checkout.html**
- As you can see, the focus order is not logical because of the `tabindex` attribute

- The `tabindex` attribute allows the developer to customize the tabbing navigation order of a document
- In **line 122**, let's remove the `tabindex="1"` attribute from the Address input field

Link Issue

- Let's start over with the Tab navigation of the page. As you can see, the **Shipping Fee Breakdown link** and the **Continue to Checkout button** does not receive keyboard focus.
- Links need the `href` attribute to receive focus.

Button Issue

- Buttons need to be structured with `<button>` to receive focus
- **Always use native HTML element first before trying to make custom controls**
- To fix the link and button:
 - Give the 'Shipping Fee Breakdown' link a `href="#"` attribute (**line 69**)
 - Structure the 'Continue to Checkout' button with the `<button>` element (**line 222**)
- **Submission:** Copy the code that removes the `tabindex` from the Address input field and the code that fixes the 'Shipping Fee Breakdown' link and 'Continue to Checkout' button. Put it in the submission form.

Exercise 7: Form Tag

- Ensure that every form element (text field, checkbox, dropdown list, etc.) has a label and make sure that label is associated with the correct form element using the `<label>` element.
- Let's take a look at the 'Zip' input field. It has no valid and visual label, which will make it very difficult for users with disability.
- To fix this:
 - Create a `<label>` element with a `'for'` attribute referencing the ID of the 'Zip' input field (**line 155**).
- Another basic thing we can do right away is to add instruction for the form.
- In the checkout form, users of cognitive disability might not understand that they have to fill out all of the form fields.
- Let's fix this by putting in an instruction at the beginning of the form that says that all form fields are required.

- **Submission:** Copy the code that gives the 'Zip' input field a visible and valid label (**line 155**). Take a picture to verify that the code for the form instruction is at the beginning of a form. Put it in the submission form.

Learning Resources

- **WebAIM** - Web Accessibility In Mind <https://webaim.org/>
 - Provide Accessibility Training
 - Has great articles on website accessibility functionalities and features <https://webaim.org/articles/>
- **WAI** - Web Accessibility Initiative <https://www.w3.org/WAI/Resources/>
 - Develops strategies, guidelines, and resources to help make the Web accessible to people with disabilities.
- **Web Content Accessibility Guidelines** (WCAG) 2.1
 - <https://www.w3.org/TR/WCAG21/>
- **Top 80 Web Accessibility Resources & Tools**
 - <https://www.websitehostingrating.com/web-accessibility-resources-tools/>

Submission & Feedback

- Please use **this attendance link** [here](#) to submit the attendance and feedback form for this Virtual TD Workshop 3.
- Thank you for joining us today! Hope you will support our business by ordering homemade bubble tea from us hehe!