Accessiblity Part 2

1. Aside from screen readers, what other assistive technologies can be affected by use of ARIA? How?

2. What is the difference between the following attributes: `hidden`, `aria-hidden="true"` and `role="presentation"` or `role="none"`

   `role="presentation"` / `role="none"`:

   - For elements with no required children, any elements nested inside the element with `role=presentation/none` preserve their semantics.

   ```
   <h1 role="presentation"><abbr>API</abbr></h1>
   ```

   For elements with required children (such as `ul` or `table`) any required child elements nested inside the element with `role=presentation/none` also have their semantics removed.

   ```
   <ul role="tablist">
     <li role="presentation">
       <a role="tab" href="#">Tab 1</a>
     </li>
     <li role="presentation">
       <a role="tab" href="#">Tab 2</a>
     </li>
     <li role="presentation">
       <a role="tab" href="#">Tab 3</a>
     </li>
   </ul>
   ```

3. Describe instances where you might need to use `aria-live`

   `aria-live` provides the ability to mark an element as containing live data so that screen readers can read out updates as they come.

- **assertive** - will interrupt whatever it's doing to announce.
- **polite** - will announce the live region update when it next idles.
- **off** - will not read the update.

4. How would you mark-up an icon font or SVG that was for decorative purposes?

   a null (empty) `alt` text should be provided (`alt=""`)

5. Is CSS pseudo content understood by screen readers?

   NO

6. What is the purpose of the `alt` attribute for images?

   - Screen readers announce alternative text in place of images, helping users with visual or certain cognitive disabilities perceive the content and function of the images.
   - If an image fails to load or the user has blocked images, the browser will present the alternative text visually in place of the image.
   - Search engines use alternative text and factor it into their assessment of the page purpose and content.

7. `<svg>` accessibility

   <svg> + role=”img” + <title> + aria-labelledby=[id]

8. Describe the steps you take in reviewing or auditing a website or application for accessibility?

9. Describe an instance where an automated test would not flag a blatant accessibility error?

10. When should you use or recommend ARIA roles or attributes to solve an accessibility issue?

11. Describe your process for figuring out if an accessibility bug is due to a developer, browser, or assistive technology error?

12. What is the difference between `legend`, `caption` and `label` elements?

    Legend is a broad label used for a group of objects.

    Label is used for labeling specific elements.

    Captions help users to find a table and understand what it's about and decide if they want to read it.

13. Describe the purpose of heading and header elements, and how they are useful in websites and applications.

    - Organizing web pages by headings helps users get a sense of the page's organization and structure.
    - HTML <header> is a collection of introductory or navigational.It can contains other heading <h1> to <h6> elements, but other elements such as a logo, wrapped section header, search type, etc are also located in the <header> section of the HTML.

14. Describe how you'd handle managing keyboard focus within a single page web app (SPA) when changing routes.

    First we needs to make the fragment programmatically focusable to begin with, Second task is to include the JavaScript `focus()` method in the callback after the new screen loads, ensuring the fragment is focused after.

15. Name an ARIA attribute that requires either a child/parent relationship or a pairing role.

For example:

an element with the role `list` will own at least one element with the role `group` or `listitem`.

a `menu` should have at least one instance of a `menuitem`, `menuitemcheckbox`, *or* `menuitemradio`.

16. What is your understanding of “accessible name computation” and how it affects modifying the way screen readers announce certain content?

    About how to generate accessible name

17. What are some issues with modifying normal scrolling behavior? For example: infinite scrolling or scrolljacking.

    infinite scrolling:

    role=feed makes infinite scroll accessible for screen reader users.

    - keyboard-only users may not able to access the footer, when tabbing to the last element will trigger scroll.
    - Some less experienced screen reader
    - Not all screen reader support.

18. Some ARIA widgets are presently best supported on devices with physical keyboard, rather than mobile/touch interfaces. Are you aware of any widgets that would be described this way, and why?