EVALUATION REPORT FOR ASSIGNMENT: How the Web Works?

Congratulations for:

- Working with a VSCode Workspace
- Having all your assets (images, styles) organized in folders
- All images contain an alt tag with a description of the image
- Congratulations! No Error Found using the W3C CSS Validator.

Suggested Tips:

Why not adding a Table of Contents that link to each sections with smooth scrolling enabled?

Smooth scrolling anyone? https://www.w3schools.com/howto/howto_css_smooth_scroll.asp

Format the document file using VSCode (Right-click on the HTML document and select Format Document) for a more readable and better organized view. Try to write like that without using the Formatter!

Use Chrome's Lighthouse to detect A11y, SEO, performance and other issues and try to fix them and produce a page evaluated at 100% on all fronts.

For example, you may need to add a `meta` description tag:

> https://web.dev/meta-description/

Prefer lowercase names for the folders/subfolders

Example: /Styles/ > /styles/

Always run your HTML and CSS code through the W3C Validators:

Try to fix all the errors and warnings that appear on your code

> HTML: https://validator.w3.org/ CSS: https://jigsaw.w3.org/css-validator/

Use the VSCode Code Spell Checker to catch any grammatical errors, e.g.

Examples: recieves -> receives, appilication -> application, sents -> sends

Use grammarly to check for any grammatical errors too

Try commenting the main sections of your page for easily reaching out to them when you need to update them. Also works, really well with code folding. Example:

```
<!-- How the Internet Work -->
<section id="how-the-internet-works" class="section_style">
```

```
    ... CONTENT ...
</section>
```

The ul tag cannot be placed inside a p tag

REFS: Google for: "can i use a ul tag inside a p tag?"

> https://stackoverflow.com/questions/5681481/should-ol-ul-be-inside-p-or-outside#answers-header
> Spec: https://html.spec.whatwg.org/#the-p-element

- REFS: https://developer.mozilla.org/en-US/docs/Web/HTML/Element/p
- REFS: https://developer.mozilla.org/en-US/docs/Web/HTML/Element/ul
- REFS: https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Content_categories#Phrasing_content

Avoid using the br tag for adding space around elements. Use CSS (margin/padding) for that Prefer the p tag to create paragraphs instead of using the br tag which is mainly used to add a breakline/newline in a single line. Example:

From this:

```
<p>
    IP address is the unique identifier assigned to a device or domain
that connects to the internet.
    <br> When connected to the internet the IP address is what allows
computers to send and recieve information.
</p>
```

To this:

```
<p>IP address is the unique identifier assigned to a device or domain that
connects to the internet</p>
<p> When connected to the internet the IP address is what allows computers
to send and recieve information.</p>
```

Prefer Semantic HTML elements instead of the too generic div tags whenever possible, e.g.

From this:

```
<div>
    <h1>HOW THE INTERNET WORKS</h1>
</div>
```

To this:

```html
<header>
    <h1>HOW THE INTERNET WORKS</h1>
</header>
```

> READ: Semantic HTML5 Elements Explained | https://www.freecodecamp.org/news/semantic-html5-elements/

Use more descriptive names in your classes and omit semantically redundant words in them, e.g.

```html
<section class="section_style">
```

Since this class is placed on all `section` tags, you don't need the `section_` prefix. And the remaining `style` is too vague. Try something more descriptive, like `basic`, `fancy`, etc.

Also, classes are mainly used for styling, so you can omit the obvious `_style` suffix.

Accessibility:

Avoid the use of phrases such as "image of ..." or "graphic of ..."; most users will be aware that an image is present, either by visual observation or adaptive technology.

REFERENCE: See the Do's and Dont's section here: https://accessibility.umn.edu/what-you-can-do/start-7-core-skills/alternative-text

CSS

Ensure that the maximum suggested characters-per-line ratio is not being overriden (~75 characters per line)

> Solution: Use the `max-width: 75ch` CSS rule on the paragraphs;

READ: https://uxdesign.cc/building-a-design-system-where-to-start-part-4-typography-5065b8d360c#a43e

Ensure that your text has the proper font-size/line-height suggested ratio (120%~140%, even 200% is acceptable)

REFS: https://practicaltypography.com/line-spacing.html

Use the Coverage tool in Chrome to detect and remove unused CSS styles (or fix them)

Chrome Command shortcut: Control+Shift+P or Command+Shift+P (Mac), then type `Coverage`

REFS: https://www.keycdn.com/blog/remove-unused-css#how-to-remove-unused-css-manually

Example: The `.how-it-works__photo figcaption { ... }` Never gets applied

REFS: Find Unused JavaScript And CSS Code With The Coverage Tab In Chrome DevTools https://developers.google.com/web/tools/chrome-devtools/coverage

REFS: Command Menu: https://developers.google.com/web/tools/chrome-devtools/command-menu

Strive for consistency when dealing and choosing font-families. For example, h1 and h2 tags have a different sans-serif font. Try sticking to a single font-family for the headings

"Using typefaces with the same family also helps your work look cohesive and clean."

Source: 5 Typography Tips for your UI https://uxplanet.org/5-typography-tips-for-your-ui-33138fe4829b

Avoid class names that are too specific and too descriptive, e.g. .how-it-works__photo

> What will happen if so some reason we decide that the section's content will change from 'How it Works' to 'How it was Born'? You will need to change the class name to reflect the updated section's content.

Stick with something like `basic_caption` or `normal_caption`.

Run your code through an CSS Linter: http://csslint.net/

REFS: What is a Linter? https://www.linkedin.com/learning/mastering-web-developer-interview-code/quick-concepts-do-you-have-experience-linting-your-code

ul, li { color, text-align, font-family } is redundant. Since all 3 CSS rules are inherited, you only need to apply them on the parent element `ul`.

Look for the `inherited` status on the following CSS rules:

REFS: https://developer.mozilla.org/en-US/docs/Web/CSS/text-align REFS: https://developer.mozilla.org/en-US/docs/Web/CSS/color REFS: https://developer.mozilla.org/en-US/docs/Web/CSS/font-family

From this:

```css
ul,
li {
    color: whitesmoke;
    text-align: center;
    font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
}
```

To this:

```css
ul {
    color: whitesmoke;
    text-align: center;
    font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
}
```

Use a web service such as TinyPNG to compress your images.