Blibli UX Engineer Internship | Written Test

Jonathan Edmund Kusnadi

1. What is a JavaScript Framework and explain about Vue.js as one of the JavaScript Framework!
   Answer:
   A JavaScript Framework, in the context of web development, is a pre-written script that handles the implementation of common utilities that are usually tedious and repetitive to write. The framework aims to simplify the development process of developing a website, so that the developer can focus on more meaningful tasks such as building features and UI, instead of handling the underlying javascript code structure.

   VueJs is one of the popular JavaScript frameworks. Personally, I haven't explored much about the VueJs Framework, however, I have been using React (NextJs) to build Frontend websites. But similar to React, Vue provides features that simplify the task of building a frontend web application. The following is an explanation of the main features of Vue:
   - Declarative Rendering:
     Vue enables developers to render data to HTML declaratively using template syntax (like EJS). This feature simplifies the process of rendering repetitive HTML views using templates and variables.
   - Components
     Vue supports the creation of HTML components, this enables the usage of reusable HTML code using only a few lines.
   - State Management
     Vue provides a reactive data binding system, so when observed data changes, the bounded UI updates automatically.
   - Event handling
     Vue provides simple directives that let us bind UI to conditional variables from an HTML attribute (ex: v-if="condition").
   - Routing
     Vue provides routing features, so different pages can be served/rendered from their associated URL paths.
   - State Manager
     Vue also comes with a state management library called Vuex that helps with complex state logic. (similar to Redux)

2. What is the use of ellipsis?
   Answer:
   Ellipsis or represented as (...) in the context of web development is used to indicate that a certain text is truncated. For example if a text in a HTML div overflows, it can be handled by limiting the text and ending the text with an ellipsis, to indicate that it has been truncated or trimmed.

3. Explain animation properties below:

a. @keyframe
Answer:
This CSS Decorator defines the behavior of an animation from a set of keyframes. A keyframe is a sequence of an animation. We can specify styles for a specific sequence in an animation using this decorator. The sequence is usually indicated by a percentage of the sequence frames.

b. animation-name
Answer:
An animation-name is a CSS attribute that indicates the name of a `@keyframe` rule that defines an animation. It associates a HTML element with an animation sequence defined.

c. animation-duration
Answer:
An animation-duration is a CSS attribute that indicates the time or duration of a single animation cycle from start to finish.

d. animation-iteration
Answer:
An animation-iteration is a CSS attribute that determines the number of animation cycles to be completed.

e. animation-direction
Answer:
An animation-direction is a CSS attribute that specifies whether an animation should play forward, backward, alternate forward and backward, or alternate backward and forward.

4. Please explain how lazy load works in JavaScript!
Answer:
Lazy loading is the practice of delaying to load a certain resource until it is needed. This optimizes the performance of a web application since the resource used will be used more efficiently.

A common lazy load practice is when loading images. We can use the loading attribute in a HTML image attribute setting it to “lazy” so that the image only loads when the viewport reaches the image element

We can also lazy load Javascript scripts by setting the HTML element
`<script async src=”script.js”></script>`
So the script loads asynchronously prioritizing the HTML parsing process.

5. Mention at least 5 git commands and describe each function of them!
Answer:
    1. `git init`
    This git command initialize a new git repository by creating a `.git` file that tracks changes in the current directory.

2. `git clone <repository-url>`
   This git command creates a copy from a remote git repository. It creates a new folder with it's associated data.
3. `git add <file(s)>`
   After adding changes to the git repository, we stage the changes using this git command. We can specify which files that we want to commit.
4. `git commit -m "commit message"`
   This git command commits the previously staged files. It is mandatory to add a commit message that describes the committed changes.
5. `git push -u origin <branch-name>`
   This git command pushes commits to the associated branch in an associated remote repository. This command also binds the local branch to the associated remote branch. If the local branch has been bound, using `git push` will do the same task.
6. `git switch <branch>`
   This git command enables us to switch between branches in a local repository. If we have multiple branches, this command will be handy.
7. `git fetch`
   This git command fetches changes or commits from the remote git repository.
8. `git pull`
   Similar to `git fetch` this command fetches changes or commits from the remote git repository, but it also merges the changes to the local repository branch.
9. `git status`
   This git command is used to see the status of the current git branch. If there are unstaged changes, staged changes, commits to be pulled from the remote repository, etc. Personally, I use this command the most to make sure that I don't make accidental modifications to the branch and prevent conflicts to the remote repository.