Godfrin
Louis

eFrei
PARIS PANTHÉON-ASSAS UNIVERSITÉ

Interface Development and Design

TP3 - VueJs

Professor : NIMAGA, Mahamadou

Godfrin
Louis

Table des matières

Godfrin
Louis

Questions:

Question 1:

> **Question 1**: That is the main difference between local installation and global installation of packages with npm? What kind of packages do you generally install locally? What kind is generally installed globally?

With local installation using npm, the packages are in the project's node_modules folder. It's only available for that specific project.

With global installation using npm, the package are on the system, making it available for all projects and from the command line.

Packages that are specific to a project, like libraries, dependencies, or frameworks are often locally installed.

Other things like command-line tools or utilities to use across projects are globally installed.

Question 2:

> **Question 2**: Webpack is internally used by the Vue CLI. Why is it required to deal with both multiple JavaScript files and special extensions like `.vue`?

Webpack is a module bundler that helps by bundling all these files into a single, optimized file. By doing so, that reduce the number of HTTP requests, and improving load times.

Webpack have a option to process and convert .vue files into standard JavaScript files. S the browsers can understand the files.

Webpack makes it easier to manage projects, especially projects with different types of files by bundling and converting files.

Godfrin
Louis

Question 3:

> **Question 3**: What is the role of babel and how `browserslist` may configure its output?

Babel is a JavaScript compiler, that converts modern JavaScript into a version that is compatible with older browsers. So developers can use the latest version of javascript without worring about compatibility issues with the browser.

Browserslist is a tool used to specify which browsers Babel target. It works by defining a list of supported browsers in the configuration.
Babel will only include the necessary polyfills, and transformations for the specified browsers in the configuration .

Question 4:

> **Question 4**: What is eslint and which set of rules are currently applied? The eslint configuration may be defined in a `eslint.config.js` or in `package.json` depending on the setup.

What is eslint:

eslint is a tool for identifying and fixing problems in JavaScript code.

It prevents errors by checking the code against a set of defined rules and enforces consistent coding.

Which set of rules are currently applied:

Eslint is present inside multiple files like package-lock.json but also package.json witch is definition is present inside the dependencies:
"@babel/eslint-parser": "^7.12.16",

```
"eslintConfig": {
  "root": true,
  "env": {
    "node": true
  },
  "extends": [
    "plugin:vue/vue3-essential",
    "eslint:recommended"
  ],
```

```
  "parserOptions": {
    "parser": "@babel/eslint-parser"
  },
  "rules": {}
 },
```

As we can see the “rules” are empty, meaning that no specified rules are currently applied.

However, because of the heritage of extends, some rules are implemented:

"plugin /vue3-essential" :

This plugin activates the essential rules for a Vue.js project with Vue 3.

It come with some default rules like:

- **vue/no-unused-components**: Prohibits declared but unused components.
- **vue/no-mutating-props**: Prevents direct mutation of props.

“eslint":

This is a set of “recommended” general rules for ESLint that covers basic JavaScript best practices.

Some examples include:

- no-unused-vars: Prohibits the use of variables that are never used.
- no-undef: Prevents the use of undefined variables.
- eqeqeq: Enforces the use of === and !== instead of == and !=.

Question 5:

Question 5: What is the difference between scoped and non-scoped CSS?

Scoped CSS is applied only to the component it is written in.
This means the styles will not affect other components or parts of the page.

Non-scoped is a global CSS that can affect any element on the page, not just within a particular component.

Godfrin
Louis

Question 6:

Question 6: How behaves non-prop attributes passed down to a component, when its template has a single root element? **Tips**: it is well documented by vue, but you can also try it youself by passing the `style` attribute with a straight visual effect.

In Vue.js, non-prop attributes like class or style for example are automatically passed to the root element of a component.

If the template has only one single root element, these attributes will be applied directly to it without needing to explicitly define them in the component.

Question 7:

Question 7: Analyse how works the `AsyncButton`. How the child component is aware of the returned Promise by the parent onClick handler? When is executed the callback passed to `.finally()`? Why use `.finally()` instead of `.then()`? Etc.

The AsyncButton component typically performs asynchronous tasks.
It is aware of the returned promise by waiting for the parent component's onClick handler to resolve the promise.

The finally() method is executed after the promise has been either resolved or rejected

The finally() method runs after both success and failure cases, making it ideal for actions that need to happen no matter what happen or issues that the code encounter.
While then() is only executed if the promise is successfully.

Godfrin
Louis

Question 8:

> **Question 8**: Which bug is introduced if `inheritAttrs: false` is missing or set to `true` in `AsyncButton`? Why?

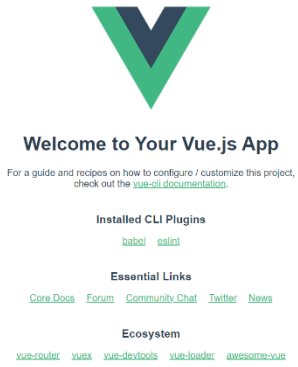
If “inheritAttrs: false” is not set in the AsyncButton component, the attributes passed to the component like class, style, etc, will be automatically applied to the root element. This might cause unexpected behavior, or styling issues if the root element is not programmed to inherit those attributes.

By setting InheritAttrs at false like that: “InheritAttrs: false”, we can manually control how, and where these attributes are applied. Help us preventing potential bugs in the layout, or behavior of the component.

Exercise:

For this part you can relate to the code inside the github repository.
I took only few screenshots while I was programming this lab.

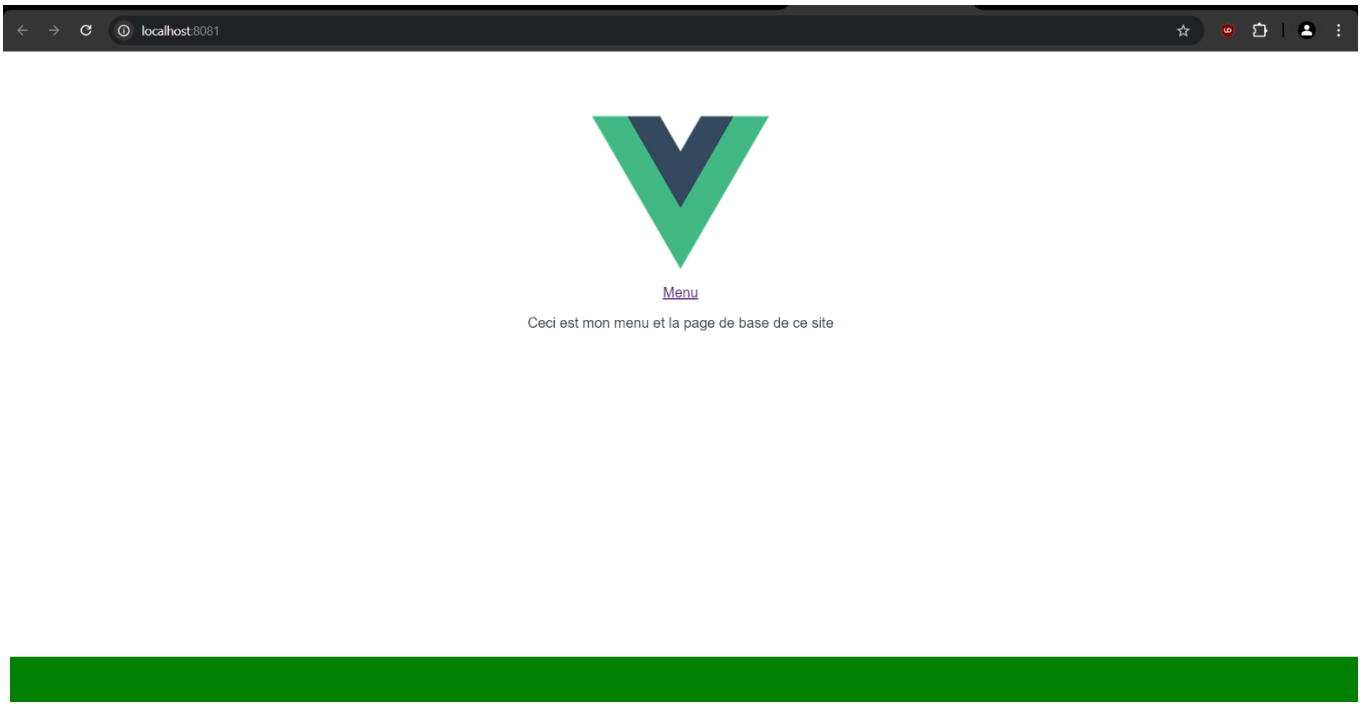
At the creation:



Exercise 4:

Godfrin
Louis

Exercise 5:



Result: