LU04b - VUE Basics

Learning objectives

1. Definition what a vue is
2. Knowing the key features of vue
3. Explaining the basic function how vue is working

Source

- Original VUE.JS
- W3School VUE

What is Vue.js?

Vue.js is a progressive JavaScript framework used for building user interfaces and single-page applications (SPAs). **It focuses on the view layer**, making it simple to integrate with other projects or libraries, while also providing powerful features for modern application development. Vue.js is known for its reactive data binding, component-based architecture, and ease of use, making it a popular choice for developers.

Vue is designed to be incrementally adoptable. Developers can start by adding it to a small part of an existing application, or they can use Vue to build entire web apps from scratch. Its lightweight nature and flexibility make it suitable for projects of all sizes.

Key Features of Vue.js

Declarative Rendering: Use Vue's syntax to bind dynamic data directly into HTML.

- Reactivity: When the data changes, the DOM updates automatically.
- Components: Modular and reusable building blocks of the interface.
- Directives: Special attributes that extend HTML functionality, like v-bind, v-if, and v-for.
- Vue CLI: A powerful tool to scaffold and manage Vue applications efficiently.

Single file Components (SFC) in Vue.js

In Vue.js, Single File Components (SFCs) are a core feature that enables developers to encapsulate the structure, style, and logic of a component **within a single .vue file**. This approach improves maintainability, readability, and scalability in modern web development by organizing all the component-related code in one place.

Structure of a Single File Component

A typical Vue SFC consists of three sections:

1. <template>: Contains the HTML structure for the component. This section defines how the component is displayed in the DOM.
2. <script>: Contains the JavaScript logic for the component, including the component's data, methods, computed properties, and lifecycle hooks.
3. <style>: Contains the CSS specific to the component, allowing you to style the component's elements. By default, these styles are global, but adding the scoped attribute makes them apply only to this component.

Example of a SFC

```
<template>
  <div class="hello">
    <h1>{{ title }}</h1>
    <p>Welcome to Vue Single File Components!</p>
    <button @click="changeTitle">Change Title</button>
  </div>
</template>
```

```
<script>
  export default {
  name: 'HelloWorld',
  data() {
    return {
      title: 'Hello, Vue!'
    };
  },
  methods: {
    changeTitle() {
      this.title = 'Title Changed!';
    }
  }
};
</script>
```

```
<style scoped>
  .hello {
    text-align: center;
    margin: 20px;
  }
  button {
    background-color: #42b983;
    color: white;
    border: none;
```

```
    padding: 10px 15px;
    cursor: pointer;
  }
  button:hover {
    background-color: #358a6c;
  }
</style>
```

Vocabulary

English	German
to streamline	beschleunigen, optimieren
incrementally adoptable	Schrittweise anwendbar
to provide	bereitstellen
to deploy	ausliefern
to enable	befähigen
to scafold	stützen