

## ALP

EXERCISE SHEET  
TEACHING ACTIVITY

Web Programming I

CURRICULAR UNIT

Exercise Sheet nº1

SHEET

M01 - Introduction to Vue.js framework

MODULE

- 1) Create a project called **F01EX01** that aims to familiarize the student with the Vue framework. Follow the next sequence of steps:
  - a. Create a folder **F01EX01**
  - b. Open the folder in VSC
  - c. Create an **index.html** file and generate the initial skeleton
  - d. Download Vue (development version) and save it in a project subfolder called **js**
  - e. Reference the vue.js file in the HTML file using a **<script>** tag
  - f. Add in the **<body>** tag the following **<div>**:

```
<div id="intro">
  <h1>{{msg}}</h1>
</div>
```

- g. Then add a **<script>** block to create an instance of Vue:

```
<script>
  const vm = new Vue({
    el: "#intro",
    data: {
      msg: "Hello Vue!"
    }
  })
</script>
```

- h. Open the browser HTML file and watch the result

2) Create a project called **F01EX02** that aims to test the link between the Vue instance and the DOM of the HTML file. Follow the next sequence of steps:

a. Add in the `<body>` tag the following `<div>`:

```
<div id="intro">
  <h2>First name: {{firstName}}</h2>
  <h2>Last name: {{lastName}}</h2>
  <h2>Age: {{age}}</h2>
  <h2>Full name: {{fullName()}}</h2>
</div>
```

b. Given the previous `<div>`, create a Vue instance that allows you to link its data through text interpolation

c. Install **devtools Chrome extension** (if not already installed)

d. Change the person's nickname on the Vue tab and check the result

3) Create a project called **F01EX03** that aims to understand the life cycle of a Vue application. Follow the next sequence of steps:

a. Add in the `<body>` tag the following `<div>`:

```
<div id="intro">
  <h2>Peso: {{weight}} kgs</h2>
</div>
```

b. Given the previous `<div>`, create a Vue instance that allows you to link its data through text interpolation

c. Add all the lifecycle functions of a Vue instance and in each include a statement such as: `console.log('[event_name]')`

d. Reload the page and check the result

e. Change the variable **weight** (via devtools) and check the result

4) Create a project called **F01EX04** that aims to understand the existing template mechanism in Vue. Follow the next sequence of steps:

a. Add the next `<script>` block:

```
const vm = new Vue({
  el: "#intro",
  data: {
    productName: "Computador Titan W599"
    productDesc: "Processador i7<br>Disco SSD 256GB<br> Monitor Samsung 21'"
    productImage: "https://www.titancomputers.com/v/vspfiles/photos/W599-7.jpg"
  }
})
```

b. Given the creation of the previous instance, create the HTML element (`<div>`) that will render the instance data. Note the type of interpolation that you should use

- c. Add a **productPrice** property to the data object with the value **1200** and adapt the HTML element to render this new information.
- d. Apply a directive so that the value appears only if it is less than **1000**. Tip: Use the **v-if** directive
- e. Change product price (via devtools) to **500** and check result
- f. Create a button that when clicked restores the original price. Tip: You must use the **v-on** directive
- g. Add a **productAvailability** property to the data object with the value **true** and adapt the HTML element to render this new information with:
  - i. the word **AVAILABLE** if the value is **true**
  - ii. the word **NOT AVAILABLE** if the value is **false**
- h. Create a button that lets you sell the product and check the result in the end