1. Data binding

- trong một đối tượng vue ta có các data muốn show ra html cho người dùng thấy thì ta cần phải ràng buộc data này vào html.

Ví dụ: ta có đối tượng này có các trường data như title, content,…

var vueInstance = new Vue({

    el:'#app',

    data: {

        title:"this is title",

        content: "this is content",

        hello: "",

    },

    methods:{

        sayHello: function(){

            this.hello = "Hello VueJs";

        }

    },

    computeds: {

    }

})

Trong Vue.js, cú pháp **{{}}** được sử dụng để binding dữ liệu. Nó là một cách để hiển thị giá trị của biến hoặc biểu thức đơn giản trong phần giao diện người dùng (UI). không thể viết một biểu thức if hay phức tạp trong {{}}

Ví dụ:

<div id='app'>

        <label>

            {{title}}

        </label>

        <p>{{content}}</p>

        <button v-on:click="sayHello()">Say Hello</button>

        <p>{{hello}}</p>

    </div>

Nhưng nếu muốn truyền data vào các thuộc tính của html như href, src…

Ta phải dùng các directive ***v-bind:<<attributes-name>>***

ví dụ:

 <div id='app'>

        <a v-bind:href="url">

            {{title}}

        </a>

        <p>{{content}}</p>

        <button v-on:click="sayHello()">Say Hello</button>

        <p>{{hello}}</p>

    </div>

var vueInstance = new Vue({

    el:'#app',

    data: {

        title:"this is title",

        content: "this is content",

        hello: "",

        url: "https://www.facebook.com/"

    },

    methods:{

        sayHello: function(){

            this.hello = "Hello VueJs";

        }

    },

    computeds: {

    }

})

1. Methods

- Methods trong vuejs instance là một hoặc nhiều phương thức trong đối tượng vuejs dùng để viết các logic xử lý dữ liệu và được gọi để phản hồi lại các sự kiện hay hành động nhất định, xử lý những tương tác của người dùng.

1. Computed