**Prop in vue js :**

Parent -> Chil

Component Parent : bind dữ liệu vào component con

 <home-page v-bind:name2="name"></home-page>

Name 2 là dữ liệu bên con sẽ nhận về

Component Chil : props để nhận dữ liệu về

Có thể khai báo kỉeue dữ liệu

Name :{

Type: ,

Required : true ( bắt buộc phải có dữ liệu truyền từ cha )

}

  props: {

      name2: ""

      },

Chil -> parent

Component chil : Emit dữ liệu lên cho parent

    changeAge () {

        this.name2 = 24;

        this.$emit('ChangName-Child',this.name2)

    }

ChangName-Child là tên sự kiện mà cha muốn lấy về

Component Cha : dung sự kiện để đổi về dữ liệu

  <home-page @ChangName-Child="name12 = $event" v-bind:name2="name"></home-page>

Tên sự kiên , tên biến muốn đặt, $event

Event bus : giao thức giữa cp khác không thông qua cha => vẫn trao đổi dữ liệu thông qua sự kiên

Đăng kí

eventBus.$emit(‘event-name’, data)

Sử dụng :

eventBus.on(‘event-name’,(data)=>

{

Use data

}

**Slot in vue js :**

**Chèn 1 đoạn html vào trong 1 component**

<template>

  <div class="title">

    {{title}}

    <slot name="slot3"></slot>

    <slot name="slot2"></slot>

  </div>

</template>

**Tại component con khai báo slot với name hoặc slot default**

    <slot-Child v-bind:title="title">

        <p slot="slot3" >hello wolrd</p>

         <p slot="slot2" >hello wolrd22</p>

    </slot-Child>

**Component cha các slot được gọi trong các thẻ html để chèn**

**Các slot viết cách nhau sẽ được gom lại cùng chỗ**

    <slot-Child v-bind:title="title">

      <p slot="slot3">hello wolrd</p>

      <p slot="slot2">hello wolrd22</p>

      <h2 slot="slot3">hello wolrd 11212</h2>

    </slot-Child>

**Ref = document.getElementbyId();**

**Tham chiếu vào một đối tượng dom bất kì khác từ một đối tượng dom**

**Vue instance :**

**Là 1 thực thể vue quản lý 1 thành phần trong dự án .**

**VD : vue instance quản lý đối tượng div #app của project**

**Life cycle :**

Before created : Chưa có gì cả ( bao gồm cả dom , event , data ,…) => k rõ làm tác dụng gì .

Created : khởi tạo thành công data , event nhưng chưa có dom => có thể để call API sẵn về để render ra trang

Before mount : khởi tạo thành công data , event nhưng chưa có dom

Mounted : đã có dom

BeforeUpdate : Khi dữ liệu **Đã** thay đổi nhưng chưa được render ra

Updated : dữ liệu **Đã** thay đổi và render ra.

Befordestroy : trước khi instance bị hủy

Computed :

Trong vue js thì bất kì một dữ liệu nào bị thay đổi thì cả template sẽ đc render lại.

Computed chỉ được gọi khi dữ liệu khai báo trong computed thay đổi

Watch :

Được chạy khi dữ liệu đăng ký theo dõi được gọi