**Props Validação**

    props: {

        nome: {

            type: String,

            default: function() {

                 return Array(10).fill(0).join(',')

            },

            required: true,

            default: 'Anônimo'

        }

    },

**Componente Filho se comunicando com o Componente Pai:** Uma das formas de se fazer isso, é o componente FILHO emitir um evento personalizado e o componente PAI trata esse evento.

Componente Filho

reiniciarNome() {

            const novo = 'Eduardo'

            // this.nome = 'Eduardo' Não usar dessa forma

            this.$emit('nomeMudou', novo)

        }

Componente Pai

<UsuarioInfo

        :nome="nome"

        @nomeMudou="nome = $event" />

Emitindo evento com mais de uma informação

Componente Filho

reiniciarNome() {

            const antigo = this.nome

            const novo = 'Eduardo'

            // this.nome = 'Eduardo' Não usar dessa forma

            this.$emit('nomeMudou', {

                novo,

                antigo

            })

        }

Componente Pai

<UsuarioInfo

        :nome="nome"

        @nomeMudou="nome = $event.novo + $event.antigo" />

**Componente Filho se comunicando com o Componente Pai: Outra forma de realizar essa comunicação é o Componente Pai passar uma função Callback para o componente Filho.**

Componente Pai

<UsuarioInfo

       :nome="nome"

       @nomeMudou="nome = $event"

       :reiniciarFn="reiniciarNome" />

<UsuarioEditar />

methods: {

     reiniciarNome() {

         this.nome = 'Eduardo'

     }

}

Componente Filho

<button @click="reiniciarFn">Reiniciar Nome (Callback)</button>

props: {

     reiniciarFn: Function

},