

VUE CHEAT SHEET



DIRECTIVAS

Elemento insertado/eliminado en función de true o, false:

```
<p v-if="inStock">{{ product }}</p>
```

```
<p v-else-if="onSale">...</p>
```

```
<p v-else>...</p>
```

Toggles display: none propiedades de CSS

```
<p v-show="showProductDetails">...</p>
```

Two-way data binding:

```
<input v-model="firstName">...</p>
```

v-model.lazy="..."

Sincroniza la entrada tras el evento "change"

v-model.number="..."

Siempre retorna un number

v-model.trim="..."

Elimina los espacios en blanco

EXPRESIONES

```
<div id="app">  
  <p> {{ product }} </p>  
  <p> {{ product + 's' }} </p>  
  <p> {{ isWorking ? 'YES' : 'NO' }} </p>  
</div>
```

BINDING

```
<a v-bind:href="url">...</a>
```

↓
Atajo v-bind
→

```
<a :href="url">...</a>
```

True o false añadirá o eliminará el atributo:

```
<button :disabled="isButtonDisabled">...</a>
```

Si isActive es verdadero, aparecerá la clase 'active':

```
<div class="isActive: is Active">...
```

VUE CHEAT SHEET



RECORRER LISTAS

```
<li v-for="item in items" :key="item.id">
  {{ item }}
</li>
```

↖ No olvides la especificar la "key"

Para acceder a la posición en el array:

```
<li v-for="(item, index) in items">...
```

Para iterar en objetos:

```
<li v-for="(value, key) in object">...
```

Utilizando v-for en un componente:

```
<Cart-product v-for="item in products"
  :product="item"
  :key="item.id">
```

PROPS

Utiliza props para pasar datos a los componentes hijos.

CUSTOM EVENTS

Utiliza eventos personalizados para pasar datos de hijos a elementos padre.

LIFECICLE HOOKS

beforeCreate
created
beforeMount
mounted

beforeUpdate
updated
beforeDestroy
destroy

EVENTOS

Llama al método addToCart del componente:

```
<button v-on:Click="addToCart">...
```

Atajo v-onClick

```
<button @Click="addToCart">...
```

Puedes pasar argumentos:

```
<button @Click="addToCart(product)">...
```

Para evitar el comportamiento por defecto
(por ejemplo, la recarga de la página):

```
<form @submit.prevent="addProduct">...
```

Ejemplo de entrada de teclado:

```
<input @keyup.enter="submit">
```

Llama a onCopy cuando se pulsa control-c:

```
<input @keyup.ctrl.c="onCopy">
```

Llama a onCopy cuando se pulsa control-c:

```
<input @keyup.ctrl.c="onCopy">
```

Modificadores para el mouse:

```
.left .right .middle
```

Modificadores para el teclado:

```
.tab .up .ctrl  
.delete .down .alt  
.esq .left .shift  
.space .right .meta
```

Sólo se dispare una vez:

```
<img @mouseover.once="showImage">...
```

Detener toda propagación de eventos

```
.stop
```