

## Cascader

Si las opciones tienen una estructura jerárquica clara, se puede utilizar Cascader para verlas y seleccionarlás.

### Uso básico

Hay dos maneras de ampliar los elementos de opción hijos.

demo Asignar el atributo `options` a un array de opciones genera un Cascader. El atributo `props.expandTrigger` define cómo se expanden las opciones hijo.

```
<div class="block">
  <span class="demonstration">Child options expand when clicked (default)</span>
  <el-cascader
    v-model="value"
    :options="options"
    @change="handleChange"></el-cascader>
</div>

<div class="block">
  <span class="demonstration">Child options expand when hovered</span>
  <el-cascader
    v-model="value"
    :options="options"
    :props="{ expandTrigger: 'hover' }"
    @change="handleChange"></el-cascader>
</div>

<script>
export default {
  data() {
    return {
      value: [],
      options: [{
        value: 'guide',
        label: 'Guide',
        children: [{
          value: 'disciplines',
          label: 'Disciplines',
          children: [{
            value: 'consistency',
            label: 'Consistency'
          }, {
            value: 'feedback',
            label: 'Feedback'
          }, {
            value: 'efficiency',
            label: 'Efficiency'
          }, {
            value: 'controllability',
            label: 'Controllability'
          }
        ]
      }, {
        value: 'feedback',
        label: 'Feedback'
      }, {
        value: 'efficiency',
        label: 'Efficiency'
      }, {
        value: 'controllability',
        label: 'Controllability'
      }
    ]
  }, {
```

```
    value: 'navigation',
    label: 'Navigation',
    children: [{
      value: 'side nav',
      label: 'Side Navigation'
    }, {
      value: 'top nav',
      label: 'Top Navigation'
    }]
  }],
}, {
  value: 'component',
  label: 'Component',
  children: [{
    value: 'basic',
    label: 'Basic',
    children: [{
      value: 'layout',
      label: 'Layout'
    }, {
      value: 'color',
      label: 'Color'
    }, {
      value: 'typography',
      label: 'Typography'
    }, {
      value: 'icon',
      label: 'Icon'
    }, {
      value: 'button',
      label: 'Button'
    }]
  }, {
    value: 'form',
    label: 'Form',
    children: [{
      value: 'radio',
      label: 'Radio'
    }, {
      value: 'checkbox',
      label: 'Checkbox'
    }, {
      value: 'input',
      label: 'Input'
    }, {
      value: 'input-number',
      label: 'InputNumber'
    }, {
      value: 'select',
      label: 'Select'
    }, {
      value: 'cascader',
```

```
    label: 'Cascader'
  }, {
    value: 'switch',
    label: 'Switch'
  }, {
    value: 'slider',
    label: 'Slider'
  }, {
    value: 'time-picker',
    label: 'TimePicker'
  }, {
    value: 'date-picker',
    label: 'DatePicker'
  }, {
    value: 'datetime-picker',
    label: 'DateTimePicker'
  }, {
    value: 'upload',
    label: 'Upload'
  }, {
    value: 'rate',
    label: 'Rate'
  }, {
    value: 'form',
    label: 'Form'
  ]
}, {
  value: 'data',
  label: 'Data',
  children: [{
    value: 'table',
    label: 'Table'
  }, {
    value: 'tag',
    label: 'Tag'
  }, {
    value: 'progress',
    label: 'Progress'
  }, {
    value: 'tree',
    label: 'Tree'
  }, {
    value: 'pagination',
    label: 'Pagination'
  }, {
    value: 'badge',
    label: 'Badge'
  ]
}, {
  value: 'notice',
  label: 'Notice',
  children: [{
```

```
    value: 'alert',
    label: 'Alert'
  }, {
    value: 'loading',
    label: 'Loading'
  }, {
    value: 'message',
    label: 'Message'
  }, {
    value: 'message-box',
    label: 'MessageBox'
  }, {
    value: 'notification',
    label: 'Notification'
  }
]
}, {
  value: 'navigation',
  label: 'Navigation',
  children: [{
    value: 'menu',
    label: 'NavMenu'
  }, {
    value: 'tabs',
    label: 'Tabs'
  }, {
    value: 'breadcrumb',
    label: 'Breadcrumb'
  }, {
    value: 'dropdown',
    label: 'Dropdown'
  }, {
    value: 'steps',
    label: 'Steps'
  }
]
}, {
  value: 'others',
  label: 'Others',
  children: [{
    value: 'dialog',
    label: 'Dialog'
  }, {
    value: 'tooltip',
    label: 'Tooltip'
  }, {
    value: 'popover',
    label: 'Popover'
  }, {
    value: 'card',
    label: 'Card'
  }, {
    value: 'carousel',
    label: 'Carousel'
  }
]
```

```

      }, {
        value: 'collapse',
        label: 'Collapse'
      }
    ]
  ]
}, {
  value: 'resource',
  label: 'Resource',
  children: [{
    value: 'axure',
    label: 'Axure Components'
  }, {
    value: 'sketch',
    label: 'Sketch Templates'
  }, {
    value: 'docs',
    label: 'Design Documentation'
  }
]
}]
};
},
methods: {
  handleChange(value) {
    console.log(value);
  }
}
};
</script>

```

...

## Opción Disabled

Deshabilite una opción estableciendo el campo `disabled` en las opciones del objeto.

:::demo En este ejemplo, el primer ítem en el array `options` tiene un campo `disabled: true`, por lo que está deshabilitado. De forma predeterminada, Cascader comprueba el campo `disabled` en cada objeto de las opciones; si está utilizando otro nombre de campo para indicar si una opción está deshabilitada, puede asignarla en el atributo `props.disabled` (consulte la tabla de la API a continuación para obtener más detalles). Y por supuesto, el nombre de campo `value`, `label` y `children` también se pueden personalizar de la misma manera.

```

<el-cascader :options="options"></el-cascader>

<script>
export default {
  data() {
    return {
      options: [{
        value: 'guide',
        label: 'Guide',

```

```
disabled: true,
children: [{
  value: 'disciplines',
  label: 'Disciplines',
  children: [{
    value: 'consistency',
    label: 'Consistency'
  }, {
    value: 'feedback',
    label: 'Feedback'
  }, {
    value: 'efficiency',
    label: 'Efficiency'
  }, {
    value: 'controllability',
    label: 'Controllability'
  }]
}, {
  value: 'navigation',
  label: 'Navigation',
  children: [{
    value: 'side nav',
    label: 'Side Navigation'
  }, {
    value: 'top nav',
    label: 'Top Navigation'
  }]
}]
}, {
  value: 'component',
  label: 'Component',
  children: [{
    value: 'basic',
    label: 'Basic',
    children: [{
      value: 'layout',
      label: 'Layout'
    }, {
      value: 'color',
      label: 'Color'
    }, {
      value: 'typography',
      label: 'Typography'
    }, {
      value: 'icon',
      label: 'Icon'
    }, {
      value: 'button',
      label: 'Button'
    }]
  }, {
    value: 'form',
```

```
label: 'Form',
children: [{
  value: 'radio',
  label: 'Radio'
}, {
  value: 'checkbox',
  label: 'Checkbox'
}, {
  value: 'input',
  label: 'Input'
}, {
  value: 'input-number',
  label: 'InputNumber'
}, {
  value: 'select',
  label: 'Select'
}, {
  value: 'cascader',
  label: 'Cascader'
}, {
  value: 'switch',
  label: 'Switch'
}, {
  value: 'slider',
  label: 'Slider'
}, {
  value: 'time-picker',
  label: 'TimePicker'
}, {
  value: 'date-picker',
  label: 'DatePicker'
}, {
  value: 'datetime-picker',
  label: 'DateTimePicker'
}, {
  value: 'upload',
  label: 'Upload'
}, {
  value: 'rate',
  label: 'Rate'
}, {
  value: 'form',
  label: 'Form'
}]
}, {
  value: 'data',
  label: 'Data',
  children: [{
    value: 'table',
    label: 'Table'
  }, {
    value: 'tag',
```

```

    label: 'Tag'
  }, {
    value: 'progress',
    label: 'Progress'
  }, {
    value: 'tree',
    label: 'Tree'
  }, {
    value: 'pagination',
    label: 'Pagination'
  }, {
    value: 'badge',
    label: 'Badge'
  }
]
}, {
  value: 'notice',
  label: 'Notice',
  children: [{
    value: 'alert',
    label: 'Alert'
  }, {
    value: 'loading',
    label: 'Loading'
  }, {
    value: 'message',
    label: 'Message'
  }, {
    value: 'message-box',
    label: 'MessageBox'
  }, {
    value: 'notification',
    label: 'Notification'
  }
]
}, {
  value: 'navigation',
  label: 'Navigation',
  children: [{
    value: 'menu',
    label: 'NavMenu'
  }, {
    value: 'tabs',
    label: 'Tabs'
  }, {
    value: 'breadcrumb',
    label: 'Breadcrumb'
  }, {
    value: 'dropdown',
    label: 'Dropdown'
  }, {
    value: 'steps',
    label: 'Steps'
  }
]
}

```



```

    }, {
      value: 'others',
      label: 'Others',
      children: [{
        value: 'dialog',
        label: 'Dialog'
      }, {
        value: 'tooltip',
        label: 'Tooltip'
      }, {
        value: 'popover',
        label: 'Popover'
      }, {
        value: 'card',
        label: 'Card'
      }, {
        value: 'carousel',
        label: 'Carousel'
      }, {
        value: 'collapse',
        label: 'Collapse'
      }
    ]
  }
], {
  value: 'resource',
  label: 'Resource',
  children: [{
    value: 'axure',
    label: 'Axure Components'
  }, {
    value: 'sketch',
    label: 'Sketch Templates'
  }, {
    value: 'docs',
    label: 'Design Documentation'
  }
]
}
];
}
};
</script>

```

:::

## Limpiable

Establezca el atributo `clearable` para `el-cascader` y aparecerá un icono de borrado cuando se seleccione y se pase el ratón por encima.

:::demo

```
<el-cascader :options="options" clearable></el-cascader>
```

```
<script>
```

```
  export default {
```

```
    data() {
```

```
      return {
```

```
        options: [{
```

```
          value: 'guide',
```

```
          label: 'Guide',
```

```
          children: [{
```

```
            value: 'disciplines',
```

```
            label: 'Disciplines',
```

```
            children: [{
```

```
              value: 'consistency',
```

```
              label: 'Consistency'
```

```
            }, {
```

```
              value: 'feedback',
```

```
              label: 'Feedback'
```

```
            }, {
```

```
              value: 'efficiency',
```

```
              label: 'Efficiency'
```

```
            }, {
```

```
              value: 'controllability',
```

```
              label: 'Controllability'
```

```
          ]]
```

```
        }, {
```

```
          value: 'navigation',
```

```
          label: 'Navigation',
```

```
          children: [{
```

```
            value: 'side nav',
```

```
            label: 'Side Navigation'
```

```
          }, {
```

```
            value: 'top nav',
```

```
            label: 'Top Navigation'
```

```
          ]]
```

```
        ]]
```

```
      }, {
```

```
        value: 'component',
```

```
        label: 'Component',
```

```
        children: [{
```

```
          value: 'basic',
```

```
          label: 'Basic',
```

```
          children: [{
```

```
            value: 'layout',
```

```
            label: 'Layout'
```

```
          }, {
```

```
            value: 'color',
```

```
            label: 'Color'
```

```
          }, {
```

```
            value: 'typography',
```

```
            label: 'Typography'
```

```
    }, {
      value: 'icon',
      label: 'Icon'
    }, {
      value: 'button',
      label: 'Button'
    }
  ]
}, {
  value: 'form',
  label: 'Form',
  children: [{
    value: 'radio',
    label: 'Radio'
  }, {
    value: 'checkbox',
    label: 'Checkbox'
  }, {
    value: 'input',
    label: 'Input'
  }, {
    value: 'input-number',
    label: 'InputNumber'
  }, {
    value: 'select',
    label: 'Select'
  }, {
    value: 'cascader',
    label: 'Cascader'
  }, {
    value: 'switch',
    label: 'Switch'
  }, {
    value: 'slider',
    label: 'Slider'
  }, {
    value: 'time-picker',
    label: 'TimePicker'
  }, {
    value: 'date-picker',
    label: 'DatePicker'
  }, {
    value: 'datetime-picker',
    label: 'DateTimePicker'
  }, {
    value: 'upload',
    label: 'Upload'
  }, {
    value: 'rate',
    label: 'Rate'
  }, {
    value: 'form',
    label: 'Form'
  }
}
```

```
    }}
  }, {
    value: 'data',
    label: 'Data',
    children: [{
      value: 'table',
      label: 'Table'
    }, {
      value: 'tag',
      label: 'Tag'
    }, {
      value: 'progress',
      label: 'Progress'
    }, {
      value: 'tree',
      label: 'Tree'
    }, {
      value: 'pagination',
      label: 'Pagination'
    }, {
      value: 'badge',
      label: 'Badge'
    }
  ]
}, {
  value: 'notice',
  label: 'Notice',
  children: [{
    value: 'alert',
    label: 'Alert'
  }, {
    value: 'loading',
    label: 'Loading'
  }, {
    value: 'message',
    label: 'Message'
  }, {
    value: 'message-box',
    label: 'MessageBox'
  }, {
    value: 'notification',
    label: 'Notification'
  }
  ]
}, {
  value: 'navigation',
  label: 'Navigation',
  children: [{
    value: 'menu',
    label: 'NavMenu'
  }, {
    value: 'tabs',
    label: 'Tabs'
  }, {

```

```

        value: 'breadcrumb',
        label: 'Breadcrumb'
    }, {
        value: 'dropdown',
        label: 'Dropdown'
    }, {
        value: 'steps',
        label: 'Steps'
    }
  ]
}, {
  value: 'others',
  label: 'Others',
  children: [{
    value: 'dialog',
    label: 'Dialog'
  }, {
    value: 'tooltip',
    label: 'Tooltip'
  }, {
    value: 'popover',
    label: 'Popover'
  }, {
    value: 'card',
    label: 'Card'
  }, {
    value: 'carousel',
    label: 'Carousel'
  }, {
    value: 'collapse',
    label: 'Collapse'
  }
  ]
}
}, {
  value: 'resource',
  label: 'Resource',
  children: [{
    value: 'axure',
    label: 'Axure Components'
  }, {
    value: 'sketch',
    label: 'Sketch Templates'
  }, {
    value: 'docs',
    label: 'Design Documentation'
  }
  ]
}
}
}
</script>

```

## Visualizar sólo el último nivel

La entrada puede mostrar sólo el último nivel en lugar de todos los niveles.

demo El atributo `show-all-levels` define si se muestran todos los niveles. Si es `false`, sólo se muestra el último nivel.

```
<el-cascader :options="options" :show-all-levels="false"></el-cascader>
<script>
  export default {
    data() {
      return {
        options: [{
          value: 'guide',
          label: 'Guide',
          children: [{
            value: 'disciplines',
            label: 'Disciplines',
            children: [{
              value: 'consistency',
              label: 'Consistency'
            }, {
              value: 'feedback',
              label: 'Feedback'
            }, {
              value: 'efficiency',
              label: 'Efficiency'
            }, {
              value: 'controllability',
              label: 'Controllability'
            }
          ]
        }, {
          value: 'navigation',
          label: 'Navigation',
          children: [{
            value: 'side nav',
            label: 'Side Navigation'
          }, {
            value: 'top nav',
            label: 'Top Navigation'
          }
        ]
      }
    ], {
      value: 'component',
      label: 'Component',
      children: [{
        value: 'basic',
        label: 'Basic',
        children: [{
          value: 'layout',
          label: 'Layout'
        }, {
```

```
    value: 'color',
    label: 'Color'
  }, {
    value: 'typography',
    label: 'Typography'
  }, {
    value: 'icon',
    label: 'Icon'
  }, {
    value: 'button',
    label: 'Button'
  }
]
}, {
  value: 'form',
  label: 'Form',
  children: [{
    value: 'radio',
    label: 'Radio'
  }, {
    value: 'checkbox',
    label: 'Checkbox'
  }, {
    value: 'input',
    label: 'Input'
  }, {
    value: 'input-number',
    label: 'InputNumber'
  }, {
    value: 'select',
    label: 'Select'
  }, {
    value: 'cascader',
    label: 'Cascader'
  }, {
    value: 'switch',
    label: 'Switch'
  }, {
    value: 'slider',
    label: 'Slider'
  }, {
    value: 'time-picker',
    label: 'TimePicker'
  }, {
    value: 'date-picker',
    label: 'DatePicker'
  }, {
    value: 'datetime-picker',
    label: 'DateTimePicker'
  }, {
    value: 'upload',
    label: 'Upload'
  }, {
```

```
    value: 'rate',
    label: 'Rate'
  }, {
    value: 'form',
    label: 'Form'
  ]
}, {
  value: 'data',
  label: 'Data',
  children: [{
    value: 'table',
    label: 'Table'
  }, {
    value: 'tag',
    label: 'Tag'
  }, {
    value: 'progress',
    label: 'Progress'
  }, {
    value: 'tree',
    label: 'Tree'
  }, {
    value: 'pagination',
    label: 'Pagination'
  }, {
    value: 'badge',
    label: 'Badge'
  ]
}, {
  value: 'notice',
  label: 'Notice',
  children: [{
    value: 'alert',
    label: 'Alert'
  }, {
    value: 'loading',
    label: 'Loading'
  }, {
    value: 'message',
    label: 'Message'
  }, {
    value: 'message-box',
    label: 'MessageBox'
  }, {
    value: 'notification',
    label: 'Notification'
  ]
}, {
  value: 'navigation',
  label: 'Navigation',
  children: [{
    value: 'menu',
```



```

        label: 'NavMenu'
    }, {
        value: 'tabs',
        label: 'Tabs'
    }, {
        value: 'breadcrumb',
        label: 'Breadcrumb'
    }, {
        value: 'dropdown',
        label: 'Dropdown'
    }, {
        value: 'steps',
        label: 'Steps'
    }
  ]
}, {
  value: 'others',
  label: 'Others',
  children: [{
    value: 'dialog',
    label: 'Dialog'
  }, {
    value: 'tooltip',
    label: 'Tooltip'
  }, {
    value: 'popover',
    label: 'Popover'
  }, {
    value: 'card',
    label: 'Card'
  }, {
    value: 'carousel',
    label: 'Carousel'
  }, {
    value: 'collapse',
    label: 'Collapse'
  }
  ]
}
], {
  value: 'resource',
  label: 'Resource',
  children: [{
    value: 'axure',
    label: 'Axure Components'
  }, {
    value: 'sketch',
    label: 'Sketch Templates'
  }, {
    value: 'docs',
    label: 'Design Documentation'
  }
  ]
}
];

```

```
    }  
  };  
</script>
```

...

## Selección múltiple

Establezca `props.multiple = true` para usar la selección múltiple.

...demo Cuando se utiliza la selección múltiple, todas las etiquetas seleccionadas se mostrarán de forma predeterminada, usted puede establecer `collapse-tags = true` para plegar las etiquetas seleccionadas.

```
<div class="block">  
  <span class="demonstration">Display all tags (default)</span>  
  <el-cascader  
    :options="options"  
    :props="props"  
    clearable></el-cascader>  
</div>  
<div class="block">  
  <span class="demonstration">Collapse tags</span>  
  <el-cascader  
    :options="options"  
    :props="props"  
    collapse-tags  
    clearable></el-cascader>  
</div>  
  
<script>  
  export default {  
    data() {  
      return {  
        props: { multiple: true },  
        options: [{  
          value: 1,  
          label: 'Asia',  
          children: [{  
            value: 2,  
            label: 'China',  
            children: [  
              { value: 3, label: 'Beijing' },  
              { value: 4, label: 'Shanghai' },  
              { value: 5, label: 'Hangzhou' }  
            ]  
          }, {  
            value: 6,  
            label: 'Japan',  
            children: [  
              { value: 7, label: 'Tokyo' },  
              { value: 8, label: 'Osaka' },  
              { value: 9, label: 'Kyoto' }  
            ]  
          }  
        ]  
      }  
    }  
  }  
}
```

```

    ]
  }, {
    value: 10,
    label: 'Korea',
    children: [
      { value: 11, label: 'Seoul' },
      { value: 12, label: 'Busan' },
      { value: 13, label: 'Taegu' }
    ]
  }]
}, {
  value: 14,
  label: 'Europe',
  children: [{
    value: 15,
    label: 'France',
    children: [
      { value: 16, label: 'Paris' },
      { value: 17, label: 'Marseille' },
      { value: 18, label: 'Lyon' }
    ]
  }], {
    value: 19,
    label: 'UK',
    children: [
      { value: 20, label: 'London' },
      { value: 21, label: 'Birmingham' },
      { value: 22, label: 'Manchester' }
    ]
  }]
}, {
  value: 23,
  label: 'North America',
  children: [{
    value: 24,
    label: 'US',
    children: [
      { value: 25, label: 'New York' },
      { value: 26, label: 'Los Angeles' },
      { value: 27, label: 'Washington' }
    ]
  }], {
    value: 28,
    label: 'Canada',
    children: [
      { value: 29, label: 'Toronto' },
      { value: 30, label: 'Montreal' },
      { value: 31, label: 'Ottawa' }
    ]
  }]
}]
};

```

```
    }  
  };  
</script>
```

...

## Seleccione cualquier nivel de opciones

En la selección única, sólo se pueden comprobar los nodos de la hoja, y en la selección múltiple, los nodos padres de la comprobación conducirán a que los nodos de la hoja se comprueben con el tiempo. Cuando se activa esta función, puede hacer que los nodos padre e hijo se desacoplen y usted puede seleccionar cualquier nivel de opciones.

demo Establezca `props.checkStrictly = true` para que el estado comprobado de un nodo no afecte a sus nodos padre y nodos hijos, y entonces podrá seleccionar cualquier nivel de opciones.

```
<div class="block">  
  <span class="demonstration">Select any level of options (Single selection)</span>  
  <el-cascader  
    :options="options"  
    :props="{ checkStrictly: true }"  
    clearable></el-cascader>  
</div>  
  
<div class="block">  
  <span class="demonstration">Select any level of options (Multiple selection)  
</span>  
  <el-cascader  
    :options="options"  
    :props="{ multiple: true, checkStrictly: true }"  
    clearable></el-cascader>  
</div>  
  
<script>  
  export default {  
    data() {  
      return {  
        options: [{  
          value: 'guide',  
          label: 'Guide',  
          children: [{  
            value: 'disciplines',  
            label: 'Disciplines',  
            children: [{  
              value: 'consistency',  
              label: 'Consistency'  
            }, {  
              value: 'feedback',  
              label: 'Feedback'  
            }, {  
              value: 'efficiency',  
              label: 'Efficiency'  
            }, {  
              value: 'consistency',  
              label: 'Consistency'  
            }, {  
              value: 'feedback',  
              label: 'Feedback'  
            }, {  
              value: 'efficiency',  
              label: 'Efficiency'  
            }  
          ]  
        }  
      ]  
    }  
  }  
}
```

```

        value: 'controllability',
        label: 'Controllability'
    ]
}, {
    value: 'navigation',
    label: 'Navigation',
    children: [{
        value: 'side nav',
        label: 'Side Navigation'
    }, {
        value: 'top nav',
        label: 'Top Navigation'
    }]
}]
}, {
    value: 'component',
    label: 'Component',
    children: [{
        value: 'basic',
        label: 'Basic',
        children: [{
            value: 'layout',
            label: 'Layout'
        }, {
            value: 'color',
            label: 'Color'
        }, {
            value: 'typography',
            label: 'Typography'
        }, {
            value: 'icon',
            label: 'Icon'
        }, {
            value: 'button',
            label: 'Button'
        }]
    }, {
        value: 'form',
        label: 'Form',
        children: [{
            value: 'radio',
            label: 'Radio'
        }, {
            value: 'checkbox',
            label: 'Checkbox'
        }, {
            value: 'input',
            label: 'Input'
        }, {
            value: 'input-number',
            label: 'InputNumber'
        }, {

```

```
    value: 'select',
    label: 'Select'
  }, {
    value: 'cascader',
    label: 'Cascader'
  }, {
    value: 'switch',
    label: 'Switch'
  }, {
    value: 'slider',
    label: 'Slider'
  }, {
    value: 'time-picker',
    label: 'TimePicker'
  }, {
    value: 'date-picker',
    label: 'DatePicker'
  }, {
    value: 'datetime-picker',
    label: 'DateTimePicker'
  }, {
    value: 'upload',
    label: 'Upload'
  }, {
    value: 'rate',
    label: 'Rate'
  }, {
    value: 'form',
    label: 'Form'
  }
]
}, {
  value: 'data',
  label: 'Data',
  children: [{
    value: 'table',
    label: 'Table'
  }, {
    value: 'tag',
    label: 'Tag'
  }, {
    value: 'progress',
    label: 'Progress'
  }, {
    value: 'tree',
    label: 'Tree'
  }, {
    value: 'pagination',
    label: 'Pagination'
  }, {
    value: 'badge',
    label: 'Badge'
  }
]
```

```
}, {
  value: 'notice',
  label: 'Notice',
  children: [{
    value: 'alert',
    label: 'Alert'
  }, {
    value: 'loading',
    label: 'Loading'
  }, {
    value: 'message',
    label: 'Message'
  }, {
    value: 'message-box',
    label: 'MessageBox'
  }, {
    value: 'notification',
    label: 'Notification'
  }]
}, {
  value: 'navigation',
  label: 'Navigation',
  children: [{
    value: 'menu',
    label: 'NavMenu'
  }, {
    value: 'tabs',
    label: 'Tabs'
  }, {
    value: 'breadcrumb',
    label: 'Breadcrumb'
  }, {
    value: 'dropdown',
    label: 'Dropdown'
  }, {
    value: 'steps',
    label: 'Steps'
  }]
}, {
  value: 'others',
  label: 'Others',
  children: [{
    value: 'dialog',
    label: 'Dialog'
  }, {
    value: 'tooltip',
    label: 'Tooltip'
  }, {
    value: 'popover',
    label: 'Popover'
  }, {
    value: 'card',
```

```

      label: 'Card'
    }, {
      value: 'carousel',
      label: 'Carousel'
    }, {
      value: 'collapse',
      label: 'Collapse'
    }
  ]
]
}, {
  value: 'resource',
  label: 'Resource',
  children: [{
    value: 'axure',
    label: 'Axure Components'
  }, {
    value: 'sketch',
    label: 'Sketch Templates'
  }, {
    value: 'docs',
    label: 'Design Documentation'
  }
]
}
];
}
};
</script>

```

...

## Carga dinámica

Carga dinámica de sus nodos hijos cuando se selecciona un nodo.

En el ejemplo establezca `lazy = true` para utilizar la carga dinámica, y deberá especificar cómo cargar la fuente de datos mediante `lazyload`. Hay dos parámetros de `lazyload`, el primer parámetro `node` es el nodo en el que se hace clic actualmente, y el `resolve` es una llamada de retorno que indica que la carga ha terminado y que debe invocarse. Para mostrar el estado del nodo con mayor precisión, puede añadir un campo `leaf` (puede ser modificado por `props.leaf`) para indicar si se trata de un nodo de hoja. De lo contrario, se deducirá verificando si tiene algún nodo hijo.

```

<el-cascader :props="props"></el-cascader>

<script>
  let id = 0;

  export default {
    data() {
      return {
        props: {
          lazy: true,

```



```

    lazyLoad (node, resolve) {
      const { level } = node;
      setTimeout(() => {
        const nodes = Array.from({ length: level + 1 })
          .map(item => ({
            value: ++id,
            label: `Option - ${id}`,
            leaf: level >= 2
          }));
        // Invoke `resolve` callback to return the child nodes data and
        indicate the loading is finished.
        resolve(nodes);
      }, 1000);
    }
  }
};
}
};
</script>

```

...

## Filtrable

Buscar y seleccionar opciones con una palabra clave.

...demo Añadir `filtrable` a `el-cascader` permite el filtrado. Cascader hará coincidir los nodos cuya etiqueta o etiqueta de padre (de acuerdo con `show-all-levels`) incluya una palabra clave de entrada. Por supuesto, puedes personalizar la lógica de búsqueda mediante el `filter-method` que acepta una función, el primer parámetro es `nodo`, el segundo es `keyword`, y necesitas devolver un valor booleano que indique si da en el blanco.

```

<div class="block">
  <span class="demonstration">Filterable (Single selection)</span>
  <el-cascader
    placeholder="Try searchingL Guide"
    :options="options"
    filterable></el-cascader>
</div>
<div class="block">
  <span class="demonstration">Filterable (Multiple selection)</span>
  <el-cascader
    placeholder="Try searchingL Guide"
    :options="options"
    :props="{ multiple: true }"
    filterable></el-cascader>
</div>

<script>
  export default {
    data() {

```

```

return {
  options: [{
    value: 'guide',
    label: 'Guide',
    children: [{
      value: 'disciplines',
      label: 'Disciplines',
      children: [{
        value: 'consistency',
        label: 'Consistency'
      }, {
        value: 'feedback',
        label: 'Feedback'
      }, {
        value: 'efficiency',
        label: 'Efficiency'
      }, {
        value: 'controllability',
        label: 'Controllability'
      }
    ]
  }, {
    value: 'navigation',
    label: 'Navigation',
    children: [{
      value: 'side nav',
      label: 'Side Navigation'
    }, {
      value: 'top nav',
      label: 'Top Navigation'
    }
  ]
}], {
  value: 'component',
  label: 'Component',
  children: [{
    value: 'basic',
    label: 'Basic',
    children: [{
      value: 'layout',
      label: 'Layout'
    }, {
      value: 'color',
      label: 'Color'
    }, {
      value: 'typography',
      label: 'Typography'
    }, {
      value: 'icon',
      label: 'Icon'
    }, {
      value: 'button',
      label: 'Button'
    }
  ]
}

```

```
    }}
  }, {
    value: 'form',
    label: 'Form',
    children: [{
      value: 'radio',
      label: 'Radio'
    }, {
      value: 'checkbox',
      label: 'Checkbox'
    }, {
      value: 'input',
      label: 'Input'
    }, {
      value: 'input-number',
      label: 'InputNumber'
    }, {
      value: 'select',
      label: 'Select'
    }, {
      value: 'cascader',
      label: 'Cascader'
    }, {
      value: 'switch',
      label: 'Switch'
    }, {
      value: 'slider',
      label: 'Slider'
    }, {
      value: 'time-picker',
      label: 'TimePicker'
    }, {
      value: 'date-picker',
      label: 'DatePicker'
    }, {
      value: 'datetime-picker',
      label: 'DateTimePicker'
    }, {
      value: 'upload',
      label: 'Upload'
    }, {
      value: 'rate',
      label: 'Rate'
    }, {
      value: 'form',
      label: 'Form'
    }]
  }, {
    value: 'data',
    label: 'Data',
    children: [{
      value: 'table',
```

```
    label: 'Table'
  }, {
    value: 'tag',
    label: 'Tag'
  }, {
    value: 'progress',
    label: 'Progress'
  }, {
    value: 'tree',
    label: 'Tree'
  }, {
    value: 'pagination',
    label: 'Pagination'
  }, {
    value: 'badge',
    label: 'Badge'
  }
]
}, {
  value: 'notice',
  label: 'Notice',
  children: [{
    value: 'alert',
    label: 'Alert'
  }, {
    value: 'loading',
    label: 'Loading'
  }, {
    value: 'message',
    label: 'Message'
  }, {
    value: 'message-box',
    label: 'MessageBox'
  }, {
    value: 'notification',
    label: 'Notification'
  }
]
}, {
  value: 'navigation',
  label: 'Navigation',
  children: [{
    value: 'menu',
    label: 'NavMenu'
  }, {
    value: 'tabs',
    label: 'Tabs'
  }, {
    value: 'breadcrumb',
    label: 'Breadcrumb'
  }, {
    value: 'dropdown',
    label: 'Dropdown'
  }, {
```

```

        value: 'steps',
        label: 'Steps'
    }
  ], {
    value: 'others',
    label: 'Others',
    children: [{
      value: 'dialog',
      label: 'Dialog'
    }, {
      value: 'tooltip',
      label: 'Tooltip'
    }, {
      value: 'popover',
      label: 'Popover'
    }, {
      value: 'card',
      label: 'Card'
    }, {
      value: 'carousel',
      label: 'Carousel'
    }, {
      value: 'collapse',
      label: 'Collapse'
    }
  ]
}
], {
  value: 'resource',
  label: 'Resource',
  children: [{
    value: 'axure',
    label: 'Axure Components'
  }, {
    value: 'sketch',
    label: 'Sketch Templates'
  }, {
    value: 'docs',
    label: 'Design Documentation'
  }
]
}
];
}
};
</script>

```

...

## Contenido de opciones personalizadas

Puede personalizar el contenido del nodo de cascada.

demo Puede personalizar el contenido del nodo del cascader mediante `scoped slot` . Tendrá acceso a `node` y `data` en el ámbito de aplicación, representando el objeto Node y los datos del nodo actual respectivamente.

```
<el-cascader :options="options">
  <template slot-scope="{ node, data }">
    <span>{{ data.label }}</span>
    <span v-if="!node.isLeaf"> ({{ data.children.length }}) </span>
  </template>
</el-cascader>

<script>
  export default {
    data() {
      return {
        options: [{
          value: 'guide',
          label: 'Guide',
          children: [{
            value: 'disciplines',
            label: 'Disciplines',
            children: [{
              value: 'consistency',
              label: 'Consistency'
            }, {
              value: 'feedback',
              label: 'Feedback'
            }, {
              value: 'efficiency',
              label: 'Efficiency'
            }, {
              value: 'controllability',
              label: 'Controllability'
            }
          ]
        }, {
          value: 'navigation',
          label: 'Navigation',
          children: [{
            value: 'side nav',
            label: 'Side Navigation'
          }, {
            value: 'top nav',
            label: 'Top Navigation'
          }
        ]
      ]
    }, {
      value: 'component',
      label: 'Component',
      children: [{
        value: 'basic',
        label: 'Basic',
        children: [{
```

```
    value: 'layout',
    label: 'Layout'
  }, {
    value: 'color',
    label: 'Color'
  }, {
    value: 'typography',
    label: 'Typography'
  }, {
    value: 'icon',
    label: 'Icon'
  }, {
    value: 'button',
    label: 'Button'
  }
]
}, {
  value: 'form',
  label: 'Form',
  children: [{
    value: 'radio',
    label: 'Radio'
  }, {
    value: 'checkbox',
    label: 'Checkbox'
  }, {
    value: 'input',
    label: 'Input'
  }, {
    value: 'input-number',
    label: 'InputNumber'
  }, {
    value: 'select',
    label: 'Select'
  }, {
    value: 'cascader',
    label: 'Cascader'
  }, {
    value: 'switch',
    label: 'Switch'
  }, {
    value: 'slider',
    label: 'Slider'
  }, {
    value: 'time-picker',
    label: 'TimePicker'
  }, {
    value: 'date-picker',
    label: 'DatePicker'
  }, {
    value: 'datetime-picker',
    label: 'DateTimePicker'
  }, {
```

```
    value: 'upload',
    label: 'Upload'
  }, {
    value: 'rate',
    label: 'Rate'
  }, {
    value: 'form',
    label: 'Form'
  }]
}, {
  value: 'data',
  label: 'Data',
  children: [{
    value: 'table',
    label: 'Table'
  }, {
    value: 'tag',
    label: 'Tag'
  }, {
    value: 'progress',
    label: 'Progress'
  }, {
    value: 'tree',
    label: 'Tree'
  }, {
    value: 'pagination',
    label: 'Pagination'
  }, {
    value: 'badge',
    label: 'Badge'
  }]
}, {
  value: 'notice',
  label: 'Notice',
  children: [{
    value: 'alert',
    label: 'Alert'
  }, {
    value: 'loading',
    label: 'Loading'
  }, {
    value: 'message',
    label: 'Message'
  }, {
    value: 'message-box',
    label: 'MessageBox'
  }, {
    value: 'notification',
    label: 'Notification'
  }]
}, {
  value: 'navigation',
```



```
    label: 'Navigation',
    children: [{
      value: 'menu',
      label: 'NavMenu'
    }, {
      value: 'tabs',
      label: 'Tabs'
    }, {
      value: 'breadcrumb',
      label: 'Breadcrumb'
    }, {
      value: 'dropdown',
      label: 'Dropdown'
    }, {
      value: 'steps',
      label: 'Steps'
    }
  ]
}, {
  value: 'others',
  label: 'Others',
  children: [{
    value: 'dialog',
    label: 'Dialog'
  }, {
    value: 'tooltip',
    label: 'Tooltip'
  }, {
    value: 'popover',
    label: 'Popover'
  }, {
    value: 'card',
    label: 'Card'
  }, {
    value: 'carousel',
    label: 'Carousel'
  }, {
    value: 'collapse',
    label: 'Collapse'
  }
  ]
}
], {
  value: 'resource',
  label: 'Resource',
  children: [{
    value: 'axure',
    label: 'Axure Components'
  }, {
    value: 'sketch',
    label: 'Sketch Templates'
  }, {
    value: 'docs',
    label: 'Design Documentation'
  }
}
```

```

    }}
  }}
}
}
}
</script>

```

...

## Panel cascader

`CascaderPanel` es el componente central de `Cascader` que tiene varias características como selección única, selección múltiple, carga dinámica, etc.

demo Al igual que `el-cascader`, puede establecer opciones alternativas mediante `options`, y habilitar otras características mediante `props`, consulte el formulario de la API a continuación para obtener más detalles.

```
<el-cascader-panel :options="options"></el-cascader-panel>
```

```

<script>
  export default {
    data() {
      return {
        options: [{
          value: 'guide',
          label: 'Guide',
          children: [{
            value: 'disciplines',
            label: 'Disciplines',
            children: [{
              value: 'consistency',
              label: 'Consistency'
            }, {
              value: 'feedback',
              label: 'Feedback'
            }, {
              value: 'efficiency',
              label: 'Efficiency'
            }, {
              value: 'controllability',
              label: 'Controllability'
            }
          ]
        }, {
          value: 'navigation',
          label: 'Navigation',
          children: [{
            value: 'side nav',
            label: 'Side Navigation'
          }, {
            value: 'top nav',
            label: 'Top Navigation'
          }
        ]
      }
    }
  }
}

```

```
    }}
  }, {
    value: 'component',
    label: 'Component',
    children: [{
      value: 'basic',
      label: 'Basic',
      children: [{
        value: 'layout',
        label: 'Layout'
      }, {
        value: 'color',
        label: 'Color'
      }, {
        value: 'typography',
        label: 'Typography'
      }, {
        value: 'icon',
        label: 'Icon'
      }, {
        value: 'button',
        label: 'Button'
      }
    ]
  }, {
    value: 'form',
    label: 'Form',
    children: [{
      value: 'radio',
      label: 'Radio'
    }, {
      value: 'checkbox',
      label: 'Checkbox'
    }, {
      value: 'input',
      label: 'Input'
    }, {
      value: 'input-number',
      label: 'InputNumber'
    }, {
      value: 'select',
      label: 'Select'
    }, {
      value: 'cascader',
      label: 'Cascader'
    }, {
      value: 'switch',
      label: 'Switch'
    }, {
      value: 'slider',
      label: 'Slider'
    }, {
      value: 'time-picker',
```

```
    label: 'TimePicker'
  }, {
    value: 'date-picker',
    label: 'DatePicker'
  }, {
    value: 'datetime-picker',
    label: 'DateTimePicker'
  }, {
    value: 'upload',
    label: 'Upload'
  }, {
    value: 'rate',
    label: 'Rate'
  }, {
    value: 'form',
    label: 'Form'
  ]
}, {
  value: 'data',
  label: 'Data',
  children: [{
    value: 'table',
    label: 'Table'
  }, {
    value: 'tag',
    label: 'Tag'
  }, {
    value: 'progress',
    label: 'Progress'
  }, {
    value: 'tree',
    label: 'Tree'
  }, {
    value: 'pagination',
    label: 'Pagination'
  }, {
    value: 'badge',
    label: 'Badge'
  ]
}, {
  value: 'notice',
  label: 'Notice',
  children: [{
    value: 'alert',
    label: 'Alert'
  }, {
    value: 'loading',
    label: 'Loading'
  }, {
    value: 'message',
    label: 'Message'
  }, {
```

```
        value: 'message-box',
        label: 'MessageBox'
    }, {
        value: 'notification',
        label: 'Notification'
    }]
}, {
    value: 'navigation',
    label: 'Navigation',
    children: [{
        value: 'menu',
        label: 'NavMenu'
    }, {
        value: 'tabs',
        label: 'Tabs'
    }, {
        value: 'breadcrumb',
        label: 'Breadcrumb'
    }, {
        value: 'dropdown',
        label: 'Dropdown'
    }, {
        value: 'steps',
        label: 'Steps'
    }]
}, {
    value: 'others',
    label: 'Others',
    children: [{
        value: 'dialog',
        label: 'Dialog'
    }, {
        value: 'tooltip',
        label: 'Tooltip'
    }, {
        value: 'popover',
        label: 'Popover'
    }, {
        value: 'card',
        label: 'Card'
    }, {
        value: 'carousel',
        label: 'Carousel'
    }, {
        value: 'collapse',
        label: 'Collapse'
    }]
}]
}, {
    value: 'resource',
    label: 'Resource',
    children: [{
```

```

        value: 'axure',
        label: 'Axure Components'
    }, {
        value: 'sketch',
        label: 'Sketch Templates'
    }, {
        value: 'docs',
        label: 'Design Documentation'
    }
  ]
}
};
</script>

```

...

## Atributos de Cascader

Atributo	Descripción	Tipo	Valores aceptados	Por defecto
value / v-model	valor ligado	-	—	—
options	datos de las opciones, las claves <code>value</code> y <code>label</code> pueden ser personalizadas por <code>Props</code> .	array	—	—
props	opciones de configuración, consulte la siguiente tabla.	object	—	—
size	tamaño del input	string	medium / small / mini	—
placeholder	placeholder del input	string	—	Select
disabled	si Cascader esta deshabilitada	boolean	—	false
clearable	si el valor seleccionado puede ser borrado	boolean	—	false
show-all-levels	si muestra todos los niveles del valor seleccionado en el input	boolean	—	true
collapse-tags	si se colapsan los tags en la selección múltiple	boolean	-	false
separator	separador de las etiquetas de las opciones	string	—	' / '
filterable	si las opciones pueden ser usadas para la búsqueda	boolean	—	—
filter-method	lógica de búsqueda personalizable. El primer parámetro es <code>node</code> , el segundo es <code>keyword</code> , y es	function(node, keyword)	-	-

	necesario devolver un valor boolean que indique si se ha tenido éxito.			
debounce	retraso en mili segundos para el tipeo de los datos de filtro	number	—	300
before-filter	hook antes de filtrar con el valor a filtrar como parámetro. Si se devuelve <code>false</code> o se devuelve una <code>Promise</code> y luego se rechaza, se abortará el filtrado.	function(value)	—	—
popper-class	nombre de clase personalizado para el menú desplegable de Cascader	string	—	—

## Eventos de Cascader

Nombre	Descripción	Parámetros
change	se desencadena cuando cambia el valor ligado.	valor
expand-change	se desencadena cuando las opciones expandidas cambian	un array de los nodos padres del nodo en expansión
blur	se desencadena cuando Cascader se desenfoca	(event: Event)
focus	se activa cuando Cascader se enfoca	(event: Event)
visible-change	se activa cuando aparece/desaparece el menú desplegable	verdadero cuando aparece, y falso de otra manera
remove-tag	se activa cuando se quita la etiqueta en modo de selección múltiple	el valor de la etiqueta que se quita

## Cascader Métodos

Método	Descripción	Parámetros
getCheckedNodes	Obtiene el array de los nodos seleccionados actualmente	(leafOnly) Si solo devuelve los nodos chequeados, por defecto es <code>false</code>

## Slots de Cascader

Nombre	Descripción
-	el contenido personalizado del nodo cascader, el parámetro es { node, data }, que son el actual objeto Node y los datos del nodo respectivamente.
empty	cuando no hay opciones coincidentes.

## Atributos del CascaderPanel

Atributos	Descripción	Tipo	Valores aceptados	Por defecto
value / v-	valor ligado	-	—	—

model				
options	datos de las opciones, las claves <code>value</code> y <code>label</code> pueden ser personalizadas por <code>Props</code> .	array	—	—
props	opciones de configuración, consulte la siguiente tabla.	object	—	—

## Eventos de CascaderPanel

Evento	Descripción	Parámetros
change	se desencadena cuando cambia el valor ligado.	valor
expand-change	se desencadena cuando las opciones expandidas cambian	un array de los nodos padres del nodo en expansión

## CascaderPanel Métodos

Método	Descripción	Parámetros
getCheckedNodes	Obtiene el array de los nodos seleccionados actualmente	(leafOnly) Si solo devuelve los nodos chequeados, por defecto es <code>false</code>
clearCheckedNodes	Limpia de nodos chequeados	-

## Slots de CascaderPanel

Nombre	Descripción
-	el contenido personalizado del nodo cascader, el parámetro es { node, data }, que son el actual objeto Node y los datos del nodo respectivamente.

## Props

Atributos	Descripción	Tipo	Valores aceptados	Por defecto
expandTrigger	modo de disparo de las opciones de ampliación	string	click / hover	'click'
multiple	si la selección múltiple esta activada	boolean	-	false
checkStrictly	si el estado verificado de un nodo no afecta a sus nodos padre e hijo	boolean	-	false
emitPath	cuando los nodos marcados cambian, si emitir o no un array de la ruta de un nodo, si es falso, sólo emite el valor del nodo.	boolean	-	true
lazy	si se deben cargar dinámicamente nodos hijo, usarlo con el atributo <code>lazyload</code> .	boolean	-	false
lazyLoad	para cargar datos de nodos hijo, sólo funciona cuando <code>lazy</code> es verdadero	function(node, resolve)	-	-



value	especificar qué clave de objeto de nodo se utiliza como valor del nodo	string	—	'value'
label	especificar qué clave de objeto de nodo se utiliza como etiqueta del nodo	string	—	'label'
children	especificar qué clave de objeto de nodo se utiliza como hijo del nodo	string	—	'children'
disabled	especificar qué clave de objeto de nodo se utiliza como nodo desactivado	string	—	'disabled'
leaf	especificar qué clave de objeto de nodo se utiliza como campo de hoja del nodo	string	—	'leaf'