

DateTimePicker 日期时间选择器

在同一个选择器里选择日期和时间

tip DateTimePicker 由 DatePicker 和 TimePicker 派生, Picker Options 或者其他选项可以参照 DatePicker 和 TimePicker。::

日期和时间点

demo 通过设置 type 属性为 datetime , 即可在同一个选择器里同时进行日期和时间的选择。快捷选项的使用方法与 Date Picker 相同。

```
<template>
  <div class="block">
    <span class="demonstration">默认</span>
    <el-date-picker
      v-model="value1"
      type="datetime"
      placeholder="选择日期时间">
    </el-date-picker>
  </div>
  <div class="block">
    <span class="demonstration">带快捷选项</span>
    <el-date-picker
      v-model="value2"
      type="datetime"
      placeholder="选择日期时间"
      align="right"
      :picker-options="pickerOptions">
    </el-date-picker>
  </div>
  <div class="block">
    <span class="demonstration">设置默认时间</span>
    <el-date-picker
      v-model="value3"
      type="datetime"
      placeholder="选择日期时间"
      default-time="12:00:00">
    </el-date-picker>
  </div>
</template>

<script>
  export default {
    data() {
      return {
        pickerOptions: {
          shortcuts: [{
            text: '今天',
            onClick(picker) {
              picker.$emit('pick', new Date());
            }
          }
        ]
      }
    }
  }
}
```

```

    }, {
      text: '昨天',
      onClick(picker) {
        const date = new Date();
        date.setTime(date.getTime() - 3600 * 1000 * 24);
        picker.$emit('pick', date);
      }
    }, {
      text: '一周前',
      onClick(picker) {
        const date = new Date();
        date.setTime(date.getTime() - 3600 * 1000 * 24 * 7);
        picker.$emit('pick', date);
      }
    }
  ]
},
value1: '',
value2: '',
value3: ''
};
}
};
</script>

```

...

日期和时间范围

::demo 设置 type 为 datetimerange 即可选择日期和时间范围

```

<template>
  <div class="block">
    <span class="demonstration">默认</span>
    <el-date-picker
      v-model="value1"
      type="datetimerange"
      range-separator="至"
      start-placeholder="开始日期"
      end-placeholder="结束日期">
    </el-date-picker>
  </div>
  <div class="block">
    <span class="demonstration">带快捷选项</span>
    <el-date-picker
      v-model="value2"
      type="datetimerange"
      :picker-options="pickerOptions"
      range-separator="至"
      start-placeholder="开始日期"
      end-placeholder="结束日期"
      align="right">
    </el-date-picker>
  </div>
</template>

```

```

    </div>
</template>

<script>
  export default {
    data() {
      return {
        pickerOptions: {
          shortcuts: [{
            text: '最近一周',
            onClick(picker) {
              const end = new Date();
              const start = new Date();
              start.setTime(start.getTime() - 3600 * 1000 * 24 * 7);
              picker.$emit('pick', [start, end]);
            }
          }, {
            text: '最近一个月',
            onClick(picker) {
              const end = new Date();
              const start = new Date();
              start.setTime(start.getTime() - 3600 * 1000 * 24 * 30);
              picker.$emit('pick', [start, end]);
            }
          }, {
            text: '最近三个月',
            onClick(picker) {
              const end = new Date();
              const start = new Date();
              start.setTime(start.getTime() - 3600 * 1000 * 24 * 90);
              picker.$emit('pick', [start, end]);
            }
          }
        ]
      },
      value1: [new Date(2000, 10, 10, 10, 10), new Date(2000, 10, 11, 10, 10)],
      value2: ''
    };
  }
};
</script>

```

...

默认的起始与结束时刻

在 `demo` 使用 `datetimerange` 进行范围选择时，在日期选择面板中选定起始与结束的日期，默认会使用该日期的 `00:00:00` 作为起始与结束的时刻；通过选项 `default-time` 可以控制选中起始与结束日期时所使用的具体时刻。`default-time` 接受一个数组，数组每项值为字符串，形如 `12:00:00`，其中第一项控制起始日期的具体时刻，第二项控制结束日期的具体时刻。

```
<template>
  <div class="block">
    <span class="demonstration">起始日期时刻为 12:00:00</span>
    <el-date-picker
      v-model="value1"
      type="datetimerange"
      start-placeholder="开始日期"
      end-placeholder="结束日期"
      :default-time="['12:00:00']">
    </el-date-picker>
  </div>
  <div class="block">
    <span class="demonstration">起始日期时刻为 12:00:00，结束日期时刻为 08:00:00</span>
    <el-date-picker
      v-model="value2"
      type="datetimerange"
      align="right"
      start-placeholder="开始日期"
      end-placeholder="结束日期"
      :default-time="['12:00:00', '08:00:00']">
    </el-date-picker>
  </div>
</template>

<script>
  export default {
    data() {
      return {
        value1: '',
        value2: ''
      };
    }
  };
</script>
```

...

Attributes

参数	说明	类型	可选值	默认值
value / v-model	绑定值	date(DateTimePicker) / array(DateTimeRangePicker)	—	—
readonly	完全只读	boolean	—	false
disabled	禁用	boolean	—	false
editable	文本框可输入	boolean	—	true
clearable	是否显示清除按钮	boolean	—	true

size	输入框尺寸	string	large, small, mini	—
placeholder	非范围选择时的占位内容	string	—	—
start-placeholder	范围选择时开始日期的占位内容	string	—	—
end-placeholder	范围选择时结束日期的占位内容	string	—	—
time-arrow-control	是否使用箭头进行时间选择	boolean	—	false
type	显示类型	string	year/month/date/week/datetime/datetimerange/daterange	date
format	显示在输入框中的格式	string	见 日期格式	YYYY-MM-HH:r
align	对齐方式	string	left, center, right	left
popper-class	DateTimePicker 下拉框的类名	string	—	—
picker-options	当前时间日期选择器特有的选项参考下表	object	—	{}
range-separator	选择范围时的分隔符	string	-	'_'
default-value	可选，选择器打开时默认显示的时间	Date	可被 <code>new Date()</code> 解析	—
default-time	选中日期后的默认具体时刻	非范围选择时：string / 范围选择时：string[]	非范围选择时：形如 <code>12:00:00</code> 的字符串；范围选择时：数组，长度为2，每项值为字符串，形如 <code>12:00:00</code> ，第一项指定开始日期的时刻，第二项指定结束日期的时刻。不指定会使用时刻 <code>00:00:00</code>	—
value-format	可选，绑定值的格式。不指定则绑定值为Date 对象	string	见 日期格式	—
name	原生属性	string	—	—
unlink-panels	在范围选择器里取消两个日	boolean	—	false

	期面板之间的 联动			
prefix-icon	自定义头部图 标的类名	string	—	el-ic date
clear-icon	自定义清空图 标的类名	string	—	el-ic circle close

Picker Options

参数	说明	类型	可 选 值	默 认 值
shortcuts	设置快捷选项，需要传入 { text, onClick } 对象用法参考 demo 或下表	Object[]	—	—
disabledDate	设置禁用状态，参数为当前日期，要求返回 Boolean	Function	—	—
cellClassName	设置日期的 className	Function(Date)	—	—
firstDayOfWeek	周起始日	Number	1 到 7	7

Shortcuts

参数	说明	类型	可 选 值	默 认 值
text	标题文本	string	—	—
onClick	选中后的回调函数，参数是 vm，可通过触发 'pick' 事件设置选择器的值。例如 vm.\$emit('pick', new Date())	function	—	—

Events

Event Name	Description	Parameters
change	用户确认选定的值时触发	组件绑定值。格式与绑定值一致，可受 value-format 控制
blur	当 input 失去焦点时触发	组件实例
focus	当 input 获得焦点时触发	组件实例

Methods

方法名	说明	参数
focus	使 input 获取焦点	—

Slots

Name	说明
range-separator	自定义分隔符