

DatePicker 日期选择器

用于选择或输入日期

选择日

以「日」为基本单位，基础的日期选择控件

demo 基本单位由 `type` 属性指定。快捷选项需配置 `picker-options` 对象中的 `shortcuts`，禁用日期通过 `disabledDate` 设置，传入函数

```
<template>
  <div class="block">
    <span class="demonstration">默认</span>
    <el-date-picker
      v-model="value1"
      type="date"
      placeholder="选择日期">
    </el-date-picker>
  </div>
  <div class="block">
    <span class="demonstration">带快捷选项</span>
    <el-date-picker
      v-model="value2"
      align="right"
      type="date"
      placeholder="选择日期"
      :picker-options="pickerOptions">
    </el-date-picker>
  </div>
</template>

<script>
export default {
  data() {
    return {
      pickerOptions: {
        disabledDate(time) {
          return time.getTime() > Date.now();
        },
        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: '',
};
}
};
</script>

```

...

其他日期单位

通过扩展基础的日期选择，可以选择周、月、年或多个日期

:::demo

```

<div class="container">
  <div class="block">
    <span class="demonstration">周</span>
    <el-date-picker
      v-model="value1"
      type="week"
      format="yyyy 第 WW 周"
      placeholder="选择周">
    </el-date-picker>
  </div>
  <div class="block">
    <span class="demonstration">月</span>
    <el-date-picker
      v-model="value2"
      type="month"
      placeholder="选择月">
    </el-date-picker>
  </div>
</div>
<div class="container">
  <div class="block">
    <span class="demonstration">年</span>
    <el-date-picker
      v-model="value3"
      type="year"
      placeholder="选择年">
    </el-date-picker>
  </div>
</div>

```

```

<div class="block">
  <span class="demonstration">多个日期</span>
  <el-date-picker
    type="dates"
    v-model="value4"
    placeholder="选择一个或多个日期">
  </el-date-picker>
</div>
</div>

<script>
  export default {
    data() {
      return {
        value1: '',
        value2: '',
        value3: '',
        value4: ''
      };
    }
  };
</script>

```

...

选择日期范围

可在一个选择器中便捷地选择一个时间范围

::demo 在选择日期范围时，默认情况下左右面板会联动。如果希望两个面板各自独立切换当前月份，可以使用 `unlink-panels` 属性解除联动。

```

<template>
  <div class="block">
    <span class="demonstration">默认</span>
    <el-date-picker
      v-model="value1"
      type="daterange"
      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="daterange"
      align="right"
      unlink-panels
      range-separator="至"
      start-placeholder="开始日期"

```

```

        end-placeholder="结束日期"
        :picker-options="pickerOptions">
    </el-date-picker>
</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: '',
    value2: ''
  },
};
</script>

```

...

选择月份范围

可在一个选择器中便捷地选择一个月份范围

...demo 在选择月份范围时，默认情况下左右面板会联动。如果希望两个面板各自独立切换当前年份，可以使用 `unlink-panels` 属性解除联动。

```

<template>
  <div class="block">
    <span class="demonstration">默认</span>
    <el-date-picker
      v-model="value1"
      type="montherange"
      range-separator="至"
      start-placeholder="开始月份"
      end-placeholder="结束月份">
    </el-date-picker>
  </div>
  <div class="block">
    <span class="demonstration">带快捷选项</span>
    <el-date-picker
      v-model="value2"
      type="montherange"
      align="right"
      unlink-panels
      range-separator="至"
      start-placeholder="开始月份"
      end-placeholder="结束月份"
      :picker-options="pickerOptions">
    </el-date-picker>
  </div>
</template>

<script>
  export default {
    data() {
      return {
        pickerOptions: {
          shortcuts: [{
            text: '本月',
            onClick(picker) {
              picker.$emit('pick', [new Date(), new Date()]);
            }
          }, {
            text: '今年至今',
            onClick(picker) {
              const end = new Date();
              const start = new Date(new Date().getFullYear(), 0);
              picker.$emit('pick', [start, end]);
            }
          }, {
            text: '最近六个月',
            onClick(picker) {
              const end = new Date();
              const start = new Date();
              start.setMonth(start.getMonth() - 6);
              picker.$emit('pick', [start, end]);
            }
          }
        ]
      }
    }
  }

```

```

    }}
  },
  value1: '',
  value2: ''
};
}
};
</script>

```

...

日期格式

使用 `format` 指定输入框的格式；使用 `value-format` 指定绑定值的格式。

默认情况下，组件接受并返回 `Date` 对象。以下为可用的格式化字符串，以 UTC 2017年1月2日 03:04:05 为例：

:::warning 请注意大小写 :::

| 格式 | 含义 | 备注 | 举例 | |-----|-----|-----|-----|-----| | `yyyy` | 年 | | 2017 | | `M` | 月 | 不补0 | 1 | | `MM` | 月 | 01 | | `W` | 周 | 仅周选择器的 `format` 可用；不补0 | 1 | | `WW` | 周 | 仅周选择器的 `format` 可用 | 01 | | `d` | 日 | 不补0 | 2 | | `dd` | 日 | 02 | | `H` | 小时 | 24小时制；不补0 | 3 | | `HH` | 小时 | 24小时制 | 03 | | `h` | 小时 | 12小时制，须和 `A` 或 `a` 使用；不补0 | 3 | | `hh` | 小时 | 12小时制，须和 `A` 或 `a` 使用 | 03 | | `m` | 分钟 | 不补0 | 4 | | `mm` | 分钟 | 04 | | `s` | 秒 | 不补0 | 5 | | `ss` | 秒 | 05 | | `A` | AM/PM | 仅 `format` 可用，大写 | AM | | `a` | am/pm | 仅 `format` 可用，小写 | am | | `timestamp` | JS时间戳 | 仅 `value-format` 可用；组件绑定值为 `number` 类型 | 1483326245000 | | `[MM]` | 不需要格式化字符 | 使用方括号标识不需要格式化的字符 (如 `[A]` `[MM]`) | MM |

:::demo

```

<template>
  <div class="block">
    <span class="demonstration">默认为 Date 对象</span>
    <div class="demonstration">值: {{ value1 }}</div>
    <el-date-picker
      v-model="value1"
      type="date"
      placeholder="选择日期"
      format="yyyy 年 MM 月 dd 日">
    </el-date-picker>
  </div>
  <div class="block">
    <span class="demonstration">使用 value-format</span>
    <div class="demonstration">值: {{ value2 }}</div>
    <el-date-picker
      v-model="value2"
      type="date"
      placeholder="选择日期"
      format="yyyy 年 MM 月 dd 日"
      value-format="yyyy-MM-dd">
    </el-date-picker>
  </div>
  <div class="block">

```

```

<span class="demonstration">时间戳</span>
<div class="demonstration">值: {{ value3 }}</div>
<el-date-picker
  v-model="value3"
  type="date"
  placeholder="选择日期"
  format="yyyy 年 MM 月 dd 日"
  value-format="timestamp">
</el-date-picker>
</div>
</template>

<script>
  export default {
    data() {
      return {
        value1: '',
        value2: '',
        value3: ''
      };
    }
  };
</script>

```

...

默认显示日期

在选择日期范围时，指定起始日期和结束日期的默认时刻。

:::demo 选择日期范围时，默认情况下，起始日期和结束日期的时间部分均为当天的 0 点 0 分 0 秒。通过 `default-time` 可以分别指定二者的具体时刻。`default-time` 接受一个数组，其中的值为形如 `12:00:00` 的字符串，第一个值控制起始日期的时刻，第二个值控制结束日期的时刻。

```

<template>
  <div class="block">
    <p>组件值: {{ value }}</p>
    <el-date-picker
      v-model="value"
      type="daterange"
      start-placeholder="开始日期"
      end-placeholder="结束日期"
      :default-time="['00:00:00', '23:59:59']">
    </el-date-picker>
  </div>
</template>

<script>
  export default {
    data() {
      return {
        value: ''
      };
    }
  };

```

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

...

Attributes

参数	说明	类型	可选值	默认值
value / v-model	绑定值	date(DatePicker) / array(DateRangePicker)	—	—
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	—	—
type	显示类型	string	year/month/date/dates/week/datetime/datetimerange/daterange/monthrange	date
format	显示在输入框中的格式	string	见 日期格式	YYYY-MM-dd
align	对齐方式	string	left, center, right	left
popper-class	DatePicker 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	范围选择时选中日期所使用的当日内具体时刻	string[]	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期面板之间的联动	boolean	—	false
prefix-icon	自定义头部图标的类名	string	—	el-icon-date
clear-icon	自定义清空图标的类名	string	—	el-icon-circle-close
validate-event	输入时是否触发表单的校验	boolean	-	true

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
onPick	选中日期后会执行的回调，只有当 daterange 或 datetimerange 才生效	Function({ maxDate, minDate })	—	—

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

事件名称	说明	回调参数
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	—