

Tabulación

Divide colecciones de datos que están relacionados pero pertenecen a diferentes tipos.

Uso básico

Tabulación básica y concisa

:::demo Tabulación provee funcionalidad de tarjeta selectiva. Por defecto, la primer pestaña es seleccionada como activa, y es posible activar cualquier pestaña estableciendo el atributo de `value`.

```
<template>
  <el-tabs v-model="activeName" @tab-click="handleClick">
    <el-tab-pane label="User" name="first">User</el-tab-pane>
    <el-tab-pane label="Config" name="second">Config</el-tab-pane>
    <el-tab-pane label="Role" name="third">Role</el-tab-pane>
    <el-tab-pane label="Task" name="fourth">Task</el-tab-pane>
  </el-tabs>
</template>
<script>
  export default {
    data() {
      return {
        activeName: 'first'
      };
    },
    methods: {
      handleClick(tab, event) {
        console.log(tab, event);
      }
    }
  };
</script>
:::
```

Estilo de Tarjeta

Pestañas diseñadas como tarjetas.

:::demo Establecer `type` a `card` para obtener una pestaña diseñada como tarjeta.

```
<template>
  <el-tabs type="card" @tab-click="handleClick">
    <el-tab-pane label="User">User</el-tab-pane>
    <el-tab-pane label="Config">Config</el-tab-pane>
    <el-tab-pane label="Role">Role</el-tab-pane>
  </el-tabs>
</template>
```

```

        <el-tab-pane label="Task">Task</el-tab-pane>
      </el-tabs>
    </template>
    <script>
      export default {
        data() {
          return {
            activeName: 'first'
          };
        },
        methods: {
          handleClick(tab, event) {
            console.log(tab, event);
          }
        }
      };
    </script>
  ...

```

Tarjeta con Bordes

Pestañas de tarjeta con bordes.

:::demo Establecer type a border-card.

```

<el-tabs type="border-card">
  <el-tab-pane label="User">User</el-tab-pane>
  <el-tab-pane label="Config">Config</el-tab-pane>
  <el-tab-pane label="Role">Role</el-tab-pane>
  <el-tab-pane label="Task">Task</el-tab-pane>
</el-tabs>
  ...

```

Posición de tabulación

Es posible usar el atributo tab-position para establecer la posición de la tabulación.

:::demo Es posible escoger entre cuatro direcciones: tabPosition="left|right|top|bottom"

```

<template>
  <el-radio-group v-model="tabPosition" style="margin-bottom: 30px;">
    <el-radio-button label="top">top</el-radio-button>
    <el-radio-button label="right">right</el-radio-button>
    <el-radio-button label="bottom">bottom</el-radio-button>
    <el-radio-button label="left">left</el-radio-button>
  </el-radio-group>

```

```

<el-tabs :tab-position="tabPosition" style="height: 200px;">
  <el-tab-pane label="User">User</el-tab-pane>
  <el-tab-pane label="Config">Config</el-tab-pane>
  <el-tab-pane label="Role">Role</el-tab-pane>
  <el-tab-pane label="Task">Task</el-tab-pane>
</el-tabs>
</template>
<script>
  export default {
    data() {
      return {
        tabPosition: 'left'
      };
    }
  };
</script>
...

```

Pestaña Personalizada

Es posible usar slots con nombre para personalizar el contenido de la etiqueta de la pestaña.

```

<el-tabs type="border-card">
  <el-tab-pane>
    <span slot="label"><i class="el-icon-date"></i> Route</span>
    Route
  </el-tab-pane>
  <el-tab-pane label="Config">Config</el-tab-pane>
  <el-tab-pane label="Role">Role</el-tab-pane>
  <el-tab-pane label="Task">Task</el-tab-pane>
</el-tabs>

```

Agregar y cerrar pestaña

Solo las pestañas de tipo tarjeta soportan adición y cierre.

```

<el-tabs v-model="editableTabsValue" type="card" editable @edit="handleTabsEdit">
  <el-tab-pane
    v-for="(item, index) in editableTabs"
    :key="item.name"
    :label="item.title"
    :name="item.name"
  >
    {{item.content}}
  </el-tab-pane>

```

```

</el-tabs>
<script>
export default {
  data() {
    return {
      editableTabsValue: '2',
      editableTabs: [{
        title: 'Tab 1',
        name: '1',
        content: 'Tab 1 content'
      }, {
        title: 'Tab 2',
        name: '2',
        content: 'Tab 2 content'
      }],
      tabIndex: 2
    }
  },
  methods: {
    handleTabsEdit(targetName, action) {
      if (action === 'add') {
        let newTabName = ++this.tabIndex + '';
        this.editableTabs.push({
          title: 'New Tab',
          name: newTabName,
          content: 'New Tab content'
        });
        this.editableTabsValue = newTabName;
      }
      if (action === 'remove') {
        let tabs = this.editableTabs;
        let activeName = this.editableTabsValue;
        if (activeName === targetName) {
          tabs.forEach((tab, index) => {
            if (tab.name === targetName) {
              let nextTab = tabs[index + 1] || tabs[index - 1];
              if (nextTab) {
                activeName = nextTab.name;
              }
            }
          });
        }

        this.editableTabsValue = activeName;
        this.editableTabs = tabs.filter(tab => tab.name !== targetName);
      }
    }
  }
}

```

```

    }
  }
}
</script>

```

Botón disparador personalizado de la nueva pestaña

```

<div style="margin-bottom: 20px;">
  <el-button
    size="small"
    @click="addTab(editableTabsValue)"
  >
    add tab
  </el-button>
</div>
<el-tabs v-model="editableTabsValue" type="card" closable @tab-remove="removeTab">
  <el-tab-pane
    v-for="(item, index) in editableTabs"
    :key="item.name"
    :label="item.title"
    :name="item.name"
  >
    {{item.content}}
  </el-tab-pane>
</el-tabs>
<script>
export default {
  data() {
    return {
      editableTabsValue: '2',
      editableTabs: [{
        title: 'Tab 1',
        name: '1',
        content: 'Tab 1 content'
      }, {
        title: 'Tab 2',
        name: '2',
        content: 'Tab 2 content'
      }],
      tabIndex: 2
    }
  },
  methods: {
    addTab(targetName) {
      let newTabName = ++this.tabIndex + '';
      this.editableTabs.push({

```

```

        title: 'New Tab',
        name: newTabName,
        content: 'New Tab content'
    });
    this.editableTabsValue = newTabName;
},
removeTab(targetName) {
    let tabs = this.editableTabs;
    let activeName = this.editableTabsValue;
    if (activeName === targetName) {
        tabs.forEach((tab, index) => {
            if (tab.name === targetName) {
                let nextTab = tabs[index + 1] || tabs[index - 1];
                if (nextTab) {
                    activeName = nextTab.name;
                }
            }
        });
    }
    this.editableTabsValue = activeName;
    this.editableTabs = tabs.filter(tab => tab.name !== targetName);
}
}
}
</script>

```

Atributos de Pestañas

Atributo	Descripción	Tipo	Valores aceptados	Por defecto
value / v-model	valor enlazado, nombre de la pestaña seleccionada	string	—	nombre de la primer pestaña
type	tipo de Pestaña	string	card/border-card	—
closable	si la Pestaña es cerrable	boolean	—	false
addable	si la Pestaña es añadible	boolean	—	false
editable	si la Pestaña es añadible y cerrable	boolean	—	false
tab- position	posición de tabulación	string	top/right/bottom/left	top

Atributo	Descripción	Tipo	Valores aceptados	Por defecto
stretch	si el ancho del tab se ajusta automáticamente a su contenedor	boolean	—	false
before-leave	función hook antes de cambiar de pestaña. Si se devuelve false o se devuelve una Promise y luego se rechaza, se evitará el cambio.	Function	{activeName, oldActiveName}	—

Eventos de Pestañas

Nombre de Evento	Descripción	Parámetros
tab-click	se lanza cuando se hace clic a una pestaña	pestaña clickeada
tab-remove	se lanza cuando se hace clic al botón tab-remove	nombre de la pestaña removida
tab-add	se lanza cuando se hace clic al botón tab-add	—
edit	se lanza cuando los botones de tab-add y/o tab-remove son cliqueados	(targetName, action)

Atributos del Tab-pane

Atributo	Descripción	Tipo	Valores Aceptados	Default
label	título de la pestaña	string	—	—
disabled	si la Tabulación está deshabilitada	boolean	—	false
name	identificador correspondiente al value de la Tabulación, representando el alias del tab-pane	string	—	número ordinal del tab-pane en la secuencia, p.ej el primer tab-pane de pestañas es '1'
closable	si el Tab es cerrable	boolean	—	false
lazy	si Tab es renderizado con lazy-load	boolean	—	false