# 表单输入绑定1.单行文本

|  |
| --- |
|  |

# 表单输入绑定2.多行文本

# 表单输入绑定3.复选框

## 1>单个复选框

|  |
| --- |
|  |

## 2>多个复选框

|  |
| --- |
|  |

# 表单绑定4.单选按钮

|  |
| --- |
|  |

# 表单绑定5:选择框select,也是有单选和多选之分

|  |
| --- |
|  |
|  |

# 演练

## 1>利用模板创建一个lesson8项目,打开App.vue文件,然后安装依赖,添加一些基本代码使用vdata和vmethod快捷键

|  |
| --- |
|  |

## 2>然后书写一些input标签,并且绑定数据,

|  |
| --- |
|  |

### 注意:

### 1.vue数据绑定有2中方式{{}}或者v-bind是单向绑定,v-model是双向绑定

### 2.给div标签绑定v-model是没有意义的

### 运行程序,效果如下

|  |
| --- |
|  |

### 此时我们修改单向绑定里面的数据发现右边并不会改变,说明模型数据没有改变

|  |
| --- |
|  |

### 然后,我们修改右边的值,左边的和下面的都会改变

|  |
| --- |
|  |

### 这就是双向绑定的效果.v-model一般和input,textarea和select等标签以及自定义标签配合使用,不要与div,ul,p,spa,table配合,无法看到效果,而且会报错,也就是只有输入类的标签才可以使用.

|  |
| --- |
|  |

## 3>项目我们给data添加一个linesMsg属性,代表多行文本.

|  |
| --- |
|  |

## 4>然后把他和一个textarea标签绑定,再添加一个p标签来显示内容

|  |
| --- |
|  |

### 运行程序,在文本域里面输入多行文本,效果如下

|  |
| --- |
|  |

## 5>然后我们在data函数中定义一个checked属性和一个checkedLangs数组属性,

|  |
| --- |
|  |

## 6>然后我们新建一个checkbox和checked属性绑定,然后在下面新建一组checkbox,然后把每一个都和这个数组绑定,注意:此时你需要给每一个checkbox添加一个value属性,因为这样做的目的其实是先获取用户选择的值

|  |
| --- |
|  |

### 运行程序,效果如下,注意看多个复选框那一栏

|  |
| --- |
|  |
|  |
|  |

## 7>也可以给单选按钮绑定模型,我们新建一个属性sex,默认是””,然后创建2个radiobutton,把name属性设置为”gender”,注意一定要一致,否则没有单选效果.只有需要给单选按钮设置value和绑定v-model

|  |
| --- |
|  |

### 运行程序,效果如下:

|  |
| --- |
|  |
|  |

### 注意:上面的多个复选框那里,我们其实可以修改一下代码,如果一个都没有选中,Selections那里不要显示[].只有选中了,才显示内容而且始终都不要显示[]

|  |
| --- |
|  |

### 运行效果:

|  |
| --- |
|  |
|  |
|  |

## 8>还可以给下拉列表select绑定模型,我们在data函数中定义一个selected属性,默认是” ”,然后在模板在添加一个下拉列表,把它的模型绑定到selected属性上面

|  |
| --- |
|  |

### 效果

|  |
| --- |
|  |
|  |
|  |
|  |

## 注意:下拉列表有一个multiple属性可以允许多选.注意此时可以按住ctrl来进行多选,这个控件比较特别,多选既可以绑定字符串,又可以绑定数组建议绑定数组.

|  |
| --- |
|  |

### 效果:

|  |
| --- |
|  |

### 我们可以修改一下代码

|  |
| --- |
|  |

### 效果

|  |
| --- |
|  |
|  |

# 扩展:v-model的使用

v-model 是Vue 中一个用于实现双向数据绑定的指令，它可以用于表单元素和自定义组件，本质上是 v-bind 和 v-on 的语法糖，使数据在视图和数据模型之间自动同步。在基础用法中，它会自动绑定表单元素的 value 属性和 input 事件；在自定义组件中，它允许指定自定义的 prop 和事件，例如 v-model:value 会绑定 props 和 update:value 事件，还可以配合修饰符如 .lazy、.number、.trim 来改变其行为。﻿

基础用法(表单元素)

在表单元素（如 <input>、<textarea>、<select>）上使用 v-model 可以实现简单的双向绑定。﻿

* **原理**：:v-model 实际上是以下代码的语法糖：
  + v-bind:value="dataValue" (或者其他属性，如 checked)
  + @input="updateDataValue($event)" (或者其他事件，如 change)

示例：

代码

<input type="text" v-model="message">

这等同于：

代码

<input type="text" :value="message" @input="message = $event.target.value">

自定义组件中的用法

在自定义组件中，v-model 也可以实现双向绑定，通常会使用 defineProps 和 defineEmits（在Vue 3 Composition API 中）。﻿

* **默认行为**：

子组件接收一个名为 modelValue 的 prop，并通过 update:modelValue 事件来通知父组件更新数据。﻿

* defineModel API (Vue 3)：

defineModel() 函数提供了一种更简洁的方式来声明带有默认值和自定义修饰符的 v-model。﻿

示例：

父组件：

代码

<MyInput v-model="parentData" />

子组件(MyInput.vue)：

代码

<script setup>  
import { defineModel } from 'vue'; // Vue 3 语法  
  
const model = defineModel(); // 获取 v-model 的值  
</script>  
  
<template>  
 <input type="text" v-model="model">  
</template>

这等同于父组件将 parentData 绑定到 MyInput 的 modelValue prop，并在 MyInput 组件内部通过 update:modelValue 事件来同步。﻿

使用修饰符

v-model 支持多种修饰符来修改其默认行为。﻿

* .lazy：将默认的 input 事件改为 change 事件，即在元素失去焦点时才进行数据同步。
* .number：强制用户输入的值转换为 Number 类型。
* .trim：自动去除输入值首尾的空白字符。

示例：

代码

<input v-model.trim.number="age" type="text">

自定义参数

还可以通过参数来创建多个 v-model 绑定到同一个组件实例。﻿

示例：

代码

<MyComponent v-model:firstName="firstName" v-model:lastName="lastName" />

在子组件中，通过 defineModel('firstName') 和 defineModel('lastName') 来分别处理。﻿