

# 题目一

---

## 题干

1、- 设计一个按钮

- 在数据源定义一个数据value，并将value渲染到页面
- 并给按钮一个事件，每次点击按钮都让value的值加10
- 当value的值等于30我们就让value的值隐藏

## 效果演示

开始：

20

按钮

当到30的时候：

按钮

30以上：

50

按钮

## 参考答案 提示：v-if和@click

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <style>

    </style>
</head>
```

```

<body>
  <div id="app">
    <div v-if="flag">{{value}}</div>
    <button @click="a">按钮</button>
  </div>
  <!-- 1. 引入vue框架 只有引入成功了 才能继续 -->
  <script src="vue.js"></script>
  <!-- 使用vue框架 -->
  <script>
    // 创建vue框架
    new Vue({
      el: "#app", // 挂载点
      data: { // 数据源
        value: 0,
        flag: true
      },
      methods: { // 事件源
        a() {
          this.value += 10 // 让这个值累加10
          if(this.value == 30){ // 如果value等于30那就让this.flag = false
            this.flag = false
          } else {
            this.flag = true // 不等于30就显示
          }
        }
      }
    })
  </script>
</body>

</html>

```

## 题目二（选做）

### 题干

2、vue来写一个登录效果。使用 vue语法实现！

补全script代码

```

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>登录页面</title>
  <style>
    * {
      margin: 0;
      padding: 0;
    }

    body {

```

```

        background-color: #f0f0f0;
        display: flex;
        justify-content: center;
        align-items: center;
        height: 100vh;
    }

    .wrapper {
        background-color: #fff;
        padding: 20px;
        border-radius: 10px;
        border: 1px solid #ccc;
        width: 250px;
    }

    .input-group {
        margin-bottom: 15px;
    }

    .input-group span {
        font-weight: 700;
        color: #333;
    }

    .input-group input {
        display: block;
        width: 90%;
        padding: 10px;
        border: 1px solid #ccc;
        border-radius: 5px;
    }

    button {
        width: 100%;
        padding: 10px;
        border: none;
        border-radius: 5px;
        background-color: #3498db;
        color: #fff;
        font-size: 16px;
        cursor: pointer;
    }

    button:hover {
        background-color: #2980b9;
    }
</style>
</head>

<body>
    <div id="app">
        <div class="wrapper">
            <div class="input-group">
                <span>用户名</span>
                <input type="text" placeholder="请输入用户名" class="username" v-
model="username">
            </div>
            <div class="input-group">

```

```

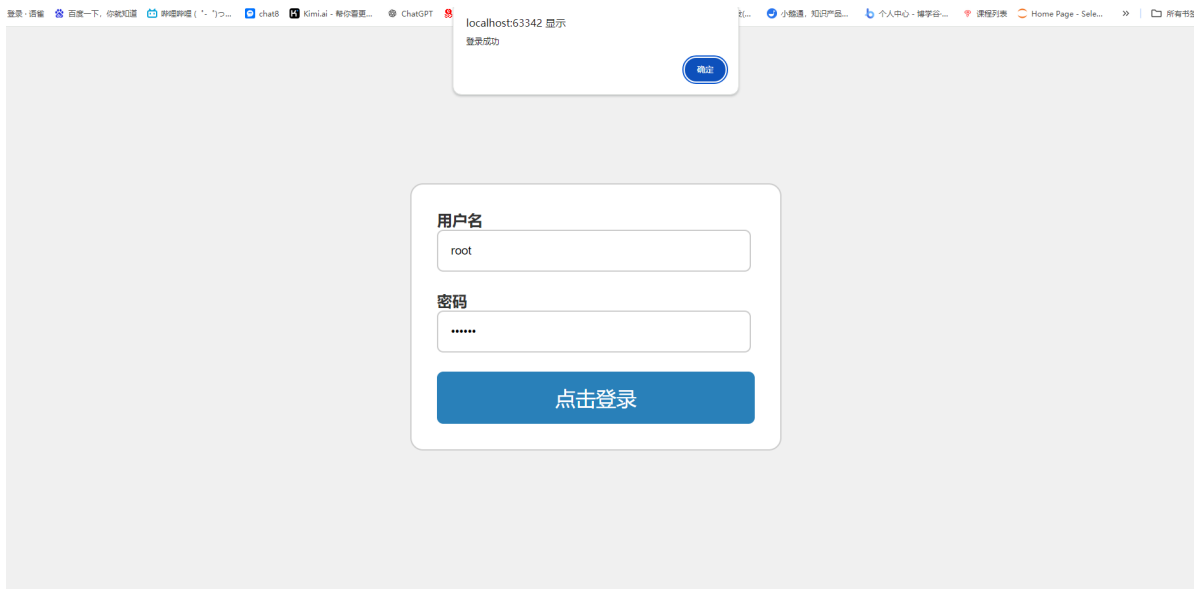
        <span>密码</span>
        <input type="password" placeholder="请输入密码" class="password" v-
model="password">
      </div>
      <button @click="login">点击登录</button>
    </div>
  </div>
<!-- 导入vue的包 -->
<script src="vue.js"></script>
<!-- 使用vue -->

<script>
  // 补全
</script>
</body>

</html>

```

## 效果演示



## 参考答案 提示：在函数login中判断输入用户名和密码是否正确

```

<script>
  new Vue({
    el: "#app",
    data: {
      username: "",
      password: ""
    },
    methods: {
      login() {
        console.log(this);
        if (this.username == "" || this.password == "") {
          alert("账号或密码不能为空")
        } else if (this.username == "root" && this.password == "123456") {
          alert("登录成功")
        } else {
          alert("账号或者密码错误")
        }
      }
    }
  })

```

```
        this.username = ""  
        this.password = ""  
    },  
    },  
    })  
</script>
```