

前言

表单是网页中最常见的视觉元素，而 **表单校验** 是表单中最常见的操作。通常情况下，会结合 **Angular/React/Vue** 等MVVM框架完成相关的表单校验功能。

本次通过纯CSS完成一个可切换的 **登录注册模块**，通过选择器实现一些看似只能由JS才能实现的效果，具体使用到如下选择器。

- **+**：相邻同胞选择器
- **~**：通用同胞选择器
- **:not()**：非指定条件的元素
- **:hover**：鼠标悬浮的元素
- **:focus**：输入聚焦的表单元素
- **:valid**：输入合法的表单元素
- **:invalid**：输入非法的表单元素
- **:checked**：选项选中的表单元素
- **:placeholder-shown**：占位显示的表单元素
- **:nth-child(n)**：元素中指定顺序索引的元素

登录注册

登录注册模块是每一个网站都可能具备的模块。本次实战需求是通过纯CSS编写一个 **登录注册模块**，当然排除 **登录** 和 **注册** 的按钮点击事件。首先需明确有哪些功能。

- 切换登录注册两个模块，可用 **~**、**:checked** 和 **nth-child(n)** 实现
- 悬浮模块导航时显示选中状态，可用 **:hover** 实现
- 判断输入框是否进入输入状态并校验内容，可用 **:focus**、**:valid** 和 **:invalid** 实现
- 判断输入框是否存在内容，可用 **+**、**:not()** 和 **placeholder-shown** 实现

总体来说，可将登录注册模块分为两部分，分别是 **模块切换** 和 **表单校验**。

模块切换

还记得第9章**选择器**如何构造这种纯CSS切换效果吗？若忘记了可回看这章，实现原理主要是结合 **<input>** 和 **<label>**。**<input>** 使用 **id** 与 **<label>** 使用 **for** 关联起来，而 **hidden** 使 **<input>** 隐藏起来，不占用页面任何位置，此时 **<label>** 放置在页面任何位置都行。

```
input:checked + div {}
input:checked ~ div {}
```

还记得第9章**选择器**的切换按钮的刹车动画吗？也搬过来使用吧，对使用刹车动画的节点声明 `transition:all 300ms cubic-bezier(.4,.4,.25,1.35)` 即可。



```
<div class="auth">
  <input id="login-btn" type="radio" name="auth" checked hidden>
  <input id="logon-btn" type="radio" name="auth" hidden>
  <div class="auth-title">
    <label for="login-btn">登录</label>
    <label for="logon-btn">注册</label>
    <em></em>
  </div>
  <div class="auth-form">
    <form>登录</form>
    <form>注册</form>
  </div>
</div>
```

```
.bruce {
  background-color: #999;
}
```

```
.auth {  
  overflow: hidden;  
  border-radius: 2px;  
  width: 320px;  
  background-color: #fff;  
  .auth-title {  
    display: flex;  
    position: relative;  
    border-bottom: 1px solid #eee;  
    height: 40px;  
    label {  
      display: flex;  
      justify-content: center;  
      align-items: center;  
      flex: 1;  
      height: 100%;  
      cursor: pointer;  
      transition: all 300ms;  
      &:hover {  
        color: #66f;  
      }  
    }  
  }  
  em {  
    position: absolute;  
    left: 0;  
    bottom: 0;  
    border-radius: 1px;  
    width: 50%;  
    height: 2px;  
    background-color: #f66;  
    transition: all 300ms cubic-bezier(.4, .4, .25, 1.35);  
  }  
}  
  .auth-form {  
    display: flex;  
    width: 200%;  
    height: 250px;
```

```
transition: all 300ms cubic-bezier(.4, .4, .25, 1.35);
form {
  flex: 1;
  padding: 20px;
}
}
}
#login-btn:checked {
  & ~ .auth-title {
    label:nth-child(1) {
      font-weight: bold;
      color: #f66;
    }
    em {
      transform: translate(0, 0);
    }
  }
  & ~ .auth-form {
    transform: translate(0, 0);
  }
}
#logon-btn:checked {
  & ~ .auth-title {
    label:nth-child(2) {
      font-weight: bold;
      color: #f66;
    }
    em {
      transform: translate(160px, 0);
    }
  }
  & ~ .auth-form {
    transform: translate(-50%, 0);
  }
}
```

表单校验

还记得第9章**选择器**的表单校验吗？通过以下几点结合能完成纯CSS表单校验，具体细节可回看这章。

完成一个完整的表单校验，需以下HTML属性和选择器搭配。

- **placeholder**：占位，在未输入内容时显示提示文本
- **pattern**：正则，在输入内容时触发正则验证
- **:valid**：作用于输入合法的表单节点
- **:invalid**：作用于输入非法的表单节点

以手机输入框为例，需满足以下HTML结构和CSS样式。

```
<input type="text" placeholder="请输入手机" pattern="^1[3456789]\d{9}$" r
```

```
input:valid {}  
input:invalid {}
```

但是存在一个问题，若直接声明 `input:valid` 和 `input:invalid`，在页面初始化后或输入框内容为空时都会触发 `:invalid`，导致表单校验还未开始就显示校验不通过的样式。为了只在输入内容时才触发 `:valid` 和 `:invalid`，可在其前面添加 `:focus`，表示在表单处于聚焦状态时才触发某些行为。

```
input:focus:valid {}  
input:focus:invalid {}
```

在输入内容时，**有内容** 和 **无内容** 可通过 `:placeholder-shown` 判断。`:placeholder-shown` 表示占位显示的表单元素，而占位不显示的表单元素可用 `:not()` 取反，再结合 `+` 带动紧随该节点的节点。

- 有内容就无占位：`:not(:placeholder-shown)`
- 无内容就有占位：`:placeholder-shown`

本次实战主要是为了将上述选择器结合起来使用而提供一种场景，所以不写太多复杂的输入框了。将登录模块的HTML结构复制一份到注册模块，哈哈！最终代码如下。



```
<div class="auth">
  <input id="login-btn" type="radio" name="auth" checked hidden>
  <input id="logon-btn" type="radio" name="auth" hidden>
  <div class="auth-title">
    <label for="login-btn">登录</label>
    <label for="logon-btn">注册</label>
    <em></em>
  </div>
  <div class="auth-form">
    <form>
      <div>
        <input type="text" placeholder="请输入手机" pattern="^1[3
        <label>手机</label>
      </div>
      <div>
        <input type="password" placeholder="请输入密码(6到20位字符)
        <label>密码</label>
      </div>
      <button type="button">登录</button>
    </form>
    <form>
      <div>
        <input type="text" placeholder="请输入手机" pattern="^1[3
        <label>手机</label>
```

```
</div>
<div>
  <input type="password" placeholder="请输入密码(6到20位字符)" />
  <label>密码</label>
</div>
<button type="button">登录</button>
</form>
</div>
</div>
```

```
.bruce {
  background-color: #999;
}
.auth {
  overflow: hidden;
  border-radius: 2px;
  width: 320px;
  background-color: #fff;
  .auth-title {
    display: flex;
    position: relative;
    border-bottom: 1px solid #eee;
    height: 40px;
    label {
      display: flex;
      justify-content: center;
      align-items: center;
      flex: 1;
      height: 100%;
      cursor: pointer;
      transition: all 300ms;
      &:hover {
        color: #66f;
      }
    }
  }
  em {
```

```
    position: absolute;
    left: 0;
    bottom: 0;
    border-radius: 1px;
    width: 50%;
    height: 2px;
    background-color: #f66;
    transition: all 300ms cubic-bezier(.4, .4, .25, 1.35);
  }
}

.auth-form {
  display: flex;
  width: 200%;
  height: 250px;
  transition: all 300ms cubic-bezier(.4, .4, .25, 1.35);
  form {
    flex: 1;
    padding: 20px;
  }
  div {
    display: flex;
    flex-direction: column-reverse;
    & + div {
      margin-top: 10px;
    }
  }
  input {
    padding: 10px;
    border: 1px solid #e9e9e9;
    border-radius: 2px;
    width: 100%;
    height: 40px;
    outline: none;
    transition: all 300ms;
    &:focus:valid {
      border-color: #09f;
    }
  }
}
```



```
&:focus:invalid {
    border-color: #f66;
}

&:not(:placeholder-shown) + label {
    height: 30px;
    opacity: 1;
    font-size: 14px;
}

label {
    overflow: hidden;
    padding: 0 10px;
    height: 0;
    opacity: 0;
    line-height: 30px;
    font-weight: bold;
    font-size: 0;
    transition: all 300ms;
}

button {
    margin-top: 10px;
    border: none;
    border-radius: 2px;
    width: 100%;
    height: 40px;
    outline: none;
    background-color: #09f;
    cursor: pointer;
    color: #fff;
    transition: all 300ms;
}

}

}

#login-btn:checked {
    & ~ .auth-title {
        label:nth-child(1) {
            font-weight: bold;
        }
    }
}
```

```
        color: #f66;
    }
    em {
        transform: translate(0, 0);
    }
}
& ~ .auth-form {
    transform: translate(0, 0);
}
}
#logon-btn:checked {
    & ~ .auth-title {
        label:nth-child(2) {
            font-weight: bold;
            color: #f66;
        }
        em {
            transform: translate(160px, 0);
        }
    }
    & ~ .auth-form {
        transform: translate(-50%, 0);
    }
}
```

☒ 在线演示: [Here](#)

☒ 在线源码: [Here](#)

留言