

<form> 元素

元素代表了表单，继承了 `HTMLFormElement` 接口。

1. HTMLFormElement 的实例属性

- `elements`: 返回一个类似数组的对象，成员是属于该表单的所有控件元素。该属性只读。
- `length`: 返回一个整数，表示属于该表单的控件数量。该属性只读。
- `name`: 字符串，表示该表单的名称。
- `method`: 字符串，表示提交给服务器时所使用的 HTTP 方法。
- `target`: 字符串，表示表单提交后，服务器返回的数据的展示位置。
- `action`: 字符串，表示表单提交数据的 URL。
- `enctype` (或 `encoding`) : 字符串，表示表单提交数据的编码方法，可能的值有 `application/x-www-form-urlencoded`、`multipart/form-data` 和 `text/plain`。
- `acceptCharset`: 字符串，表示服务器所能接受的字符编码，多个编码格式之间使用逗号或空格分隔。
- `autocomplete`: 字符串 "on" 或 "off"，表示浏览器是否要对 `<input>` 控件提供自动补全。
- `noValidate`: 布尔值，表示是否关闭表单的自动校验。

以下图登录表单为例：

 请输入账号

 请输入登录密码

登录

```
let form = document.getElementsByClassName('ant-form')[0];

form.length; // 3
form.elements; // HTMLFormControlsCollection(3) {0: input.ant-input, 1:
input.ant-input, 2: button.ant-btn, length: 3}
form.method; // "get"
form.action; // "http://39.105.131.127:32002/#/passport/login"

form.elements[0] instanceof HTMLInputElement; // true
form.elements instanceof HTMLFormControlsCollection; // true
```

`get` 属性返回的值不准确，如果没有在 `<form>` 元素上显示的设置 `method` 属性，默认返回 `get`。但我们提交这个登录表单时是使用的 `post` 方式。

`action` 属性返回的值不准确, 如果没有在 `<form>` 元素上显示的设置 `action` 属性, 默认返回当前登录页的 `location.href`。但是我这里请求地址是 `"http://39.105.131.127:32002/api/auth/login"`

2. HTMLFormElement 的实例方法

- `submit()`: 提交表单, 但是不会触发 `submit` 事件和表单的自动校验。
- `reset()`: 重置表单控件的值为默认值。
- `checkValidity()`: 如果控件能够通过自动校验, 返回 `true`, 否则返回 `false`, 同时触发 `invalid` 事件。