

ACCESSIBLE FORM INSTRUCTIONS

Overall form
instructions

Where relevant, instructions that apply to the entire form **should be provided before the `<form>` element.**

For example, indicate any **required fields**, allowed **date formats**, and **password limitations**.

```
<p>All form fields are required.</p>
```

```
<form action="#">
```

```
...
```

```
</form>
```

Inline instructions

There are times when individual form fields **need additional information** in order for users to understand how to fill them in correctly.

These instructions **need to be programmatically associated with the form control**, so that they are announced to screen readers along with the label content.

Different methods

There are **three different methods** that can be used to programmatically associate instructions with their relevant form fields.

Method 1:
Inside the label

Using this method, information can be associated with form fields by **placing it inside the `<label>` element.**

This method only works if the `<label>` element **wraps around the label content, form control and instructions.**

```
<label for="one">  
  <span class="label-content">Phone number</span>  
  <input id="one" type="text">  
    <span class="hint">Make sure to include a 10  
    digit number</span>  
</label>
```

This technique is **stable across all web browsers and assistive technologies** so is the recommended solution.

Method 2:
Using aria-describedby

Using this method, the `aria-`
`describedby` attribute can be used to
**associate instructions with form
controls.**

In this example, the `<input>` element **has**
an `aria-describedby` **attribute** with a
value of `instructions2`.

The `` element, which contains the additional instructions, **also has an ID value of instructions2**.

```
<label for="one">Phone number</label>
<input id="one" type="text"
  aria-describedby="instructions1">
<span id="instructions1">Make sure to include a
10 digit number</span>
```

This technique has **good support** across all modern web browsers and assistive technologies.

It can be used in cases where it is **not possible** to wrap the `<label>` element around the form control.

Method 3.
Using placeholder

This method involves using the `placeholder` attribute to display simple instructions **directly in the form field.**


```
<label for="one">Phone number</label>  
<input id="one" type="text"  
  placeholder="Make sure to include a 10 digit  
  number">
```

This method presents **two accessibility issues:**

1. These instructions **disappear** as soon as the user types a character in the field, or tab to the field.

For some users, especially some types of cognitive impaired users, this can present a challenge **as the instructions are no longer available.**

2. The placeholder text is often **presented in a very soft gray color** which is well below acceptable WCAG color contrast ratios.

Even though this text can be styled with CSS, many developers do not do this, so **it can be inaccessible to anyone with vision issues.**

For these reasons, the `placeholder` method **should be avoided**.

Exercise 03:

Form Instructions

Open **exercise03-instructions/
exercise3-start.html**

Review your work against **exercise03-
instructions/exercise3-finished.html**

- Move overall form instructions to the correct location.
- Solve the “name” form control by wrapping the `<label>`.
- Solve the “email” form control using the `aria-describedby` attribute.