3.3 Common form widgets

Checkbox

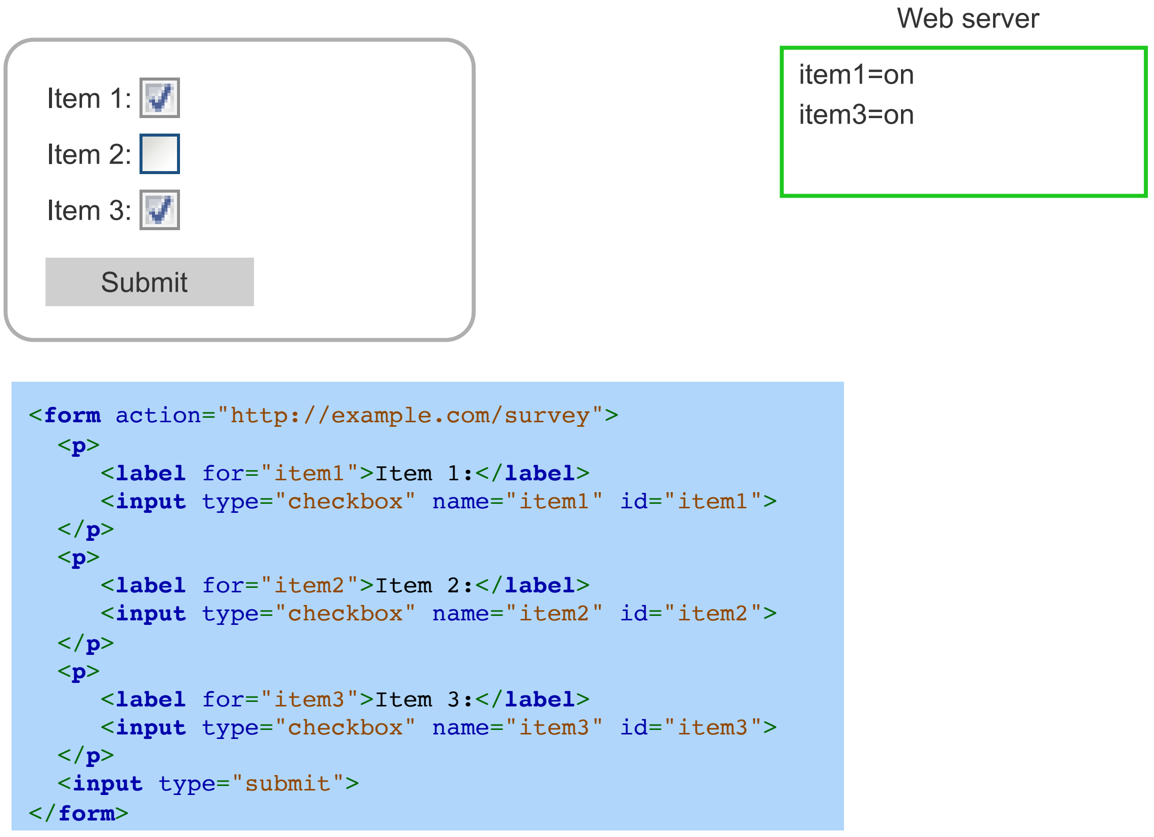
A ***checkbox*** is a widget for input elements with the `type` attribute of "checkbox", which allows users to check, or select, a value. A checkbox initially appears selected if the ***checked attribute*** is set. Ex: `<input type="checkbox" checked>` creates a checked checkbox. The `checked` attribute is an example of a boolean attribute. A ***boolean attribute*** is an attribute that is true when present and false when absent. No value must be assigned to a boolean attribute.

For each checkbox selected, the browser sends the checkbox's name and value to the server. If the value attribute is not specified, the default value of "on" is sent. If a checkbox is not selected, the browser does not send anything to the server. *A common error on the server is failing to record all checkboxes in the form as the browser doesn't report any values for checkboxes not selected by the user.*

Good practice is to use label elements with checkboxes so the user can click the label to check and uncheck the associated checkbox.

PARTICIPATION ACTIVITY | 3.3.1: Submitting checkboxes to the server.

Start □ 2x speed



```
item1=on
item3=on
```

```
<form action="http://example.com/survey">
  <p>
     <label for="item1">Item 1:</label>
     <input type="checkbox" name="item1" id="item1">
  </p>
  <p>
     <label for="item2">Item 2:</label>
     <input type="checkbox" name="item2" id="item2">
  </p>
  <p>
     <label for="item3">Item 3:</label>
     <input type="checkbox" name="item3" id="item3">
  </p>
  <input type="submit">
</form>
```

Captions

Captions

1. Each checkbox is displayed with a corresponding label.
2. The user checks Item 1 and Item 3 checkboxes and clicks the submit button.
3. For each selected checkbox, the checkbox's name and the value "on" are sent to the server.

Feedback?

PARTICIPATION ACTIVITY 3.3.2: Checkbox inputs. ✓

The following form contains a checkbox, which is initially checked. Click the submit button to view the form data sent to the server. Try adding another checkbox with the name "throw_party".

```
<form action="https://wp.zybooks.com/form-viewer.php" target="_b
   <p>
      <label for="birthday_today">Birthday today?</label>
      <input type="checkbox" id="birthday_today" name="birthday_
   </p>

   <p>
      <input type="submit" value="Submit">
   </p>
</form>

```

Render webpage Reset code

Your webpage

Birthday today? ☑

Submit

Expected webpage

Birthday today? ☑

Throw party? ☑

Submit

▼ View solution

Explain

```
--- START FILE: HTML ---

<form action="https://wp.zybooks.com/form-viewer.php"
target="_blank" method="GET">
   <p>
      <label for="birthday_today">Birthday today?</label>
      <input type="checkbox" id="birthday_today"
name="birthday_today" checked>
   </p>

   <p>
      <label for="throw_party">Throw party?</label>
      <input type="checkbox" id="throw_party"
name="throw_party" checked>
   </p>

   <p>
      <input type="submit" value="Submit">
   </p>
</form>


--- END FILE: HTML ---
```

Feedback?

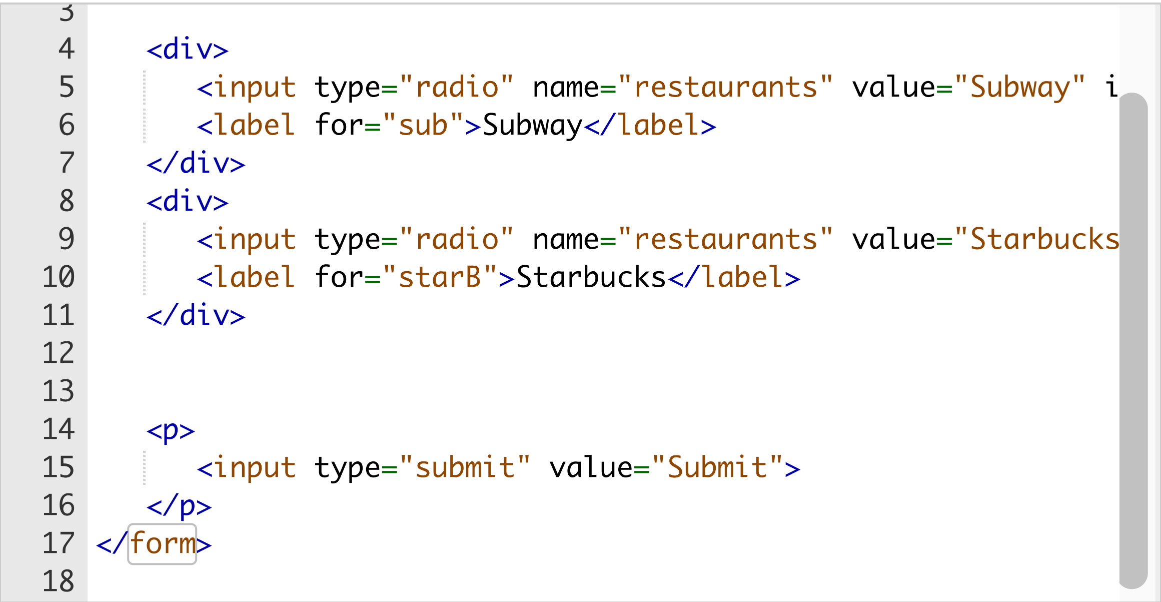
Radio button

A ***radio button*** is a widget for input elements with the `type` attribute of "radio", which allows users to select exactly one value from possibly many values. The web browser groups radio buttons together with the same `name` attribute, where each possible value in a group has an associated input. When submitting a form, the browser sends the selected radio button's `name` and `value` attribute. Ex: If the radio button `<input type="radio" name="movie" value="ET">` is selected, "movie=ET" is sent to the server.

The main difference between a radio button and checkbox is that only one radio button in a group can be selected, while any number of checkboxes can be selected.

PARTICIPATION ACTIVITY 3.3.3: Radio buttons. ✓

Try adding another radio option for the restaurants group with a value of Pizza Hut. Click the submit button to view the form data sent to server. Then, try adding another set of radio buttons to match the expected rendered HTML.



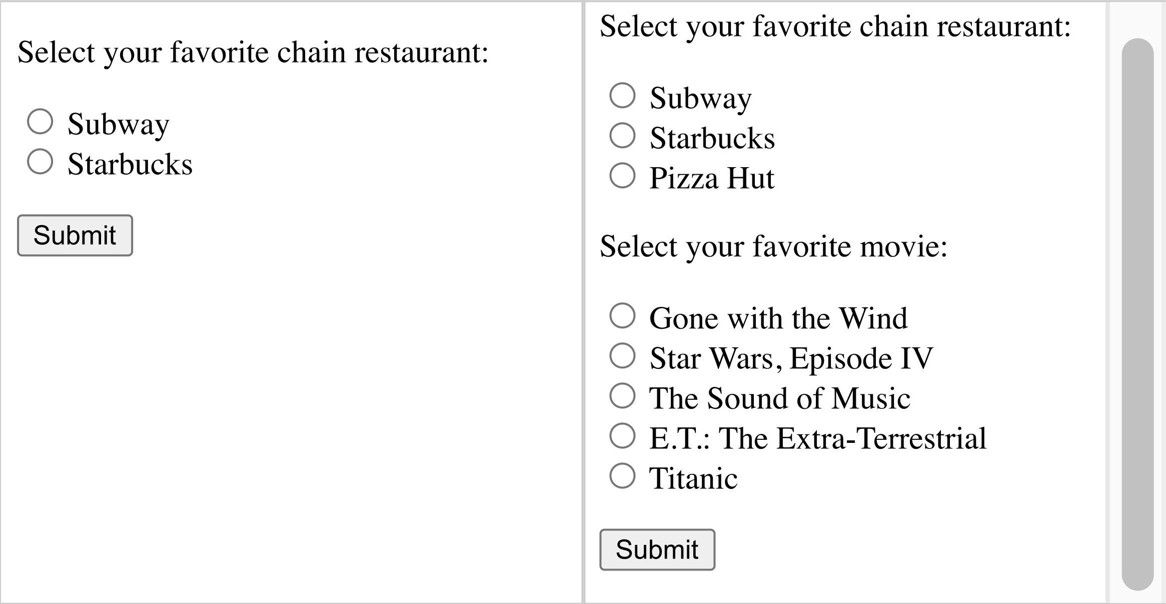
```
   <div>
      <input type="radio" name="restaurants" value="Subway" i
      <label for="sub">Subway</label>
   </div>
   <div>
      <input type="radio" name="restaurants" value="Starbucks
      <label for="starB">Starbucks</label>
   </div>


   <p>
      <input type="submit" value="Submit">
   </p>
</form>

```

Render webpage   Reset code

Your webpage **Expected webpage**



Select your favorite chain restaurant:

○ Subway
○ Starbucks

Submit

Select your favorite chain restaurant:

○ Subway
○ Starbucks
○ Pizza Hut

Select your favorite movie:

○ Gone with the Wind
○ Star Wars, Episode IV
○ The Sound of Music
○ E.T.: The Extra-Terrestrial
○ Titanic

Submit

▼ View solution

Explain

```
--- START FILE: HTML ---

<form action="https://wp.zybooks.com/form-viewer.php"
target="_blank" method="GET">
   <p>Select your favorite chain restaurant:</p>

   <div>
```

```
      <input type="radio" name="restaurants" value="Subway"
id="sub">
      <label for="sub">Subway</label>
   </div>
   <div>
      <input type="radio" name="restaurants"
value="Starbucks" id="starB">
      <label for="starB">Starbucks</label>
   </div>
   <div>
      <input type="radio" name="restaurants"
value="PizzaHut" id="pizzaH">
      <label for="pizzaH">Pizza Hut</label>
   </div>

   <p>Select your favorite movie:</p>
   <div>
      <input type="radio" name="movies"
value="GoneWithTheWind" id="gwtw">
      <label for="gwtw">Gone with the Wind</label>
   </div>
   <div>
      <input type="radio" name="movies"
value="StarWarsEpisodeIV" id="sw">
      <label for="sw">Star Wars, Episode IV</label>
   </div>
   <div>
      <input type="radio" name="movies"
value="TheSoundOfMusic" id="tsom">
      <label for="tsom">The Sound of Music</label>
   </div>
   <div>
      <input type="radio" name="movies" value="ET" id="et" >
      <label for="et">E.T.: The Extra-Terrestrial</label>
   </div>
   <div>
      <input type="radio" name="movies" value="Titanic"
id="titanic" >
      <label for="titanic">Titanic</label>
   </div>

   <p>
      <input type="submit" value="Submit">
   </p>
</form>
```

```
--- END FILE: HTML ---
```

Feedback?

PARTICIPATION ACTIVITY | 3.3.4: Checkboxes and radio buttons. ✓

Refer to the HTML below.

```
<form action="form-processor" method="POST">
   <p>Select a flavor:</p>
   <div>
      <input type="radio" name="flavor" value="1" checked>
      <label for="choc">Chocolate</label>
   </div>
   <div>
      <input type="radio" name="flavor" value="2">
      <label for="van">Vanilla</label>
   </div>
   <div>
      <input type="radio" name="flavor" value="3">
      <label for="straw">Strawberry</label>
   </div>

   <p>Select toppings:</p>
   <div>
      <input type="checkbox" name="topping-whip">
      <label for="whip">Whipped Cream</label>
   </div>
   <div>
      <input type="checkbox" name="topping-nuts" checked>
      <label for="nuts">Nuts</label>
   </div>

   <p>
      <input type="submit" value="Order">
   </p>
</form>
```

1) If the user clicks Order without changing anything in the form, how many items will be sent to the server?

2

Check Show answer

Correct

2

Radio buttons and checkboxes that have a `checked` attribute are selected/checked by default. So the browser will send the two items: flavor=1 and topping-nuts=on.

✓

2) What is the maximum number of items that can be sent to the server?

3

Correct

3

The `name` attribute for each radio button is the same, so only one radio button can be selected. Both

✓

Check Show answer

checkboxes can be checked. If one radio button is selected and both checkboxes are checked, 3 items are sent.

3) What is the minimum number of items that will be sent to the server?

1

Check Show answer

Correct ✓

1

Once a radio button is selected from a group, the browser requires one radio button always be selected from that group. A radio button cannot be unselected like a checkbox. So, the browser will minimally send the selected flavor.

4) The labels are not correctly associated with each input element. Which attribute should be added to each input element to correctly associate the labels with the input elements?

id

Check Show answer

Correct ✓

id

The `<label>` tag's `for` attribute must match the associated element's `id` attribute. Ex: `<input id="van" ...>` for the label `<label for="van">`.

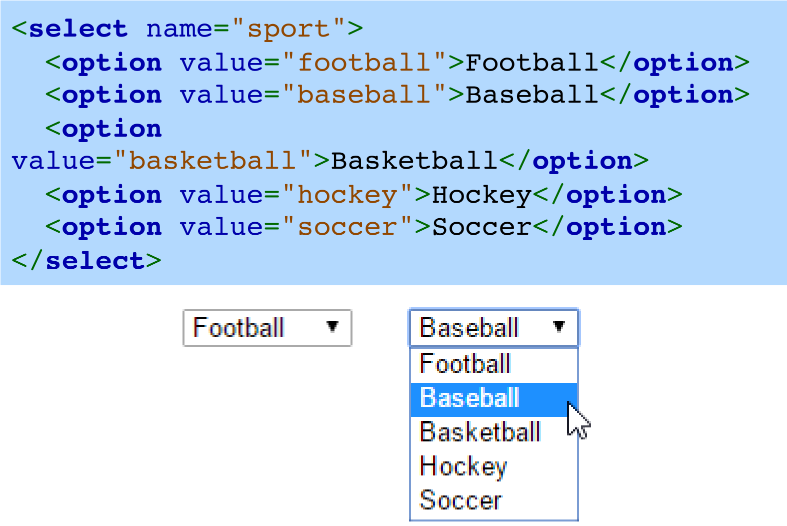
Feedback?

Drop-down menu

The ***<select>*** opening and closing tags create a ***drop-down menu*** (or ***drop-down list***), which allows users to select one of several predefined values. The ***<option>*** opening and closing tags create a value, or option, the user can select within a drop-down menu. When the user is not interacting with the menu, the drop-down menu usually displays the selected option.

The difference between a drop-down menu and a radio button widget is that the drop-down menu only displays the options when interacting with the user, while a radio button widget always displays all options.

Figure 3.3.1: Drop-down menu's default appearance (left) and when selecting an option (right).



```
<select name="sport">
  <option value="football">Football</option>
  <option value="baseball">Baseball</option>
  <option
value="basketball">Basketball</option>
  <option value="hockey">Hockey</option>
  <option value="soccer">Soccer</option>
</select>
```

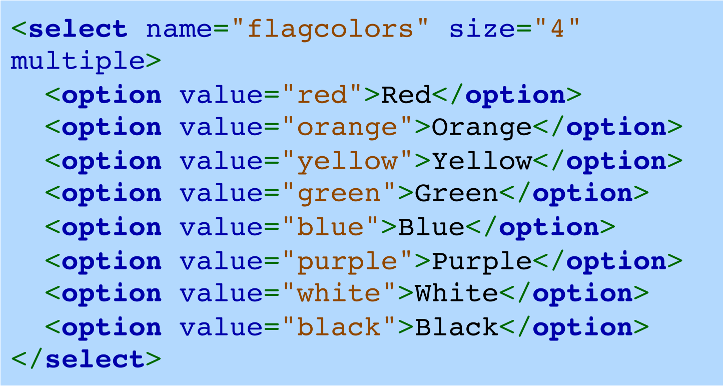
Feedback?

List box

A ***list box*** widget is created by specifying a size with the select element's `size` attribute. Ex: `<select size="4">` creates a list box that shows four options at a time. If the list box contains more than `size` options, the browser adds a vertical scrollbar so the user can scroll through the list of options.

The `multiple` attribute allows the user to select multiple options. On Windows, the user must hold down the control (Ctrl) button to select multiple options, and on a Mac, the user must hold down the command button. *Many users are unaware of how to choose multiple options from a list box, so good practice is to use checkboxes instead.*

Figure 3.3.2: List box that allows multiple options to be selected.



```
<select name="flagcolors" size="4"
multiple>
  <option value="red">Red</option>
  <option value="orange">Orange</option>
  <option value="yellow">Yellow</option>
  <option value="green">Green</option>
  <option value="blue">Blue</option>
  <option value="purple">Purple</option>
  <option value="white">White</option>
  <option value="black">Black</option>
</select>
```

Feedback?

Buttons

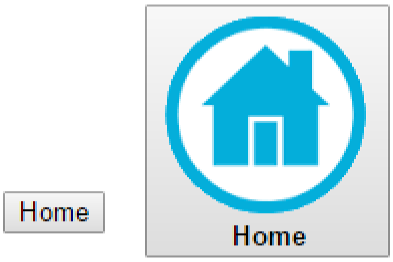
A ***button*** widget can be created using the ***<button>*** opening and closing tags or with `<input type="button">`. The `<button>` element allows text and images to be displayed in a button, but an `<input>` button only allows text.

The `<button>` element has a `type` attribute that can be set to various values like "button" or "submit". The "button" type is typically used with JavaScript to perform an action when clicked. The "submit" type creates a submit button for a form. *If the `type` attribute is not specified, different browsers may choose different default types, so good practice is to always specify the type.*

Figure 3.3.3: HTML buttons.

```
<input type="button" value="Home">

<button type="button">
  <img src="home.png" alt="home">
<br>
  <strong>Home</strong>
</button>
```

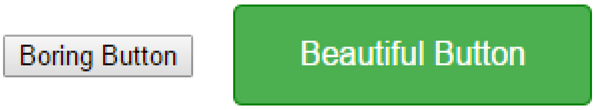


Feedback?

Styling widgets

The default look of a form's widgets may differ depending on the browser and operating system. Developers use CSS to give widgets a more uniform look or *to increase the widgets' visual appeal.*

The image on the left shows the default button in Chrome. The button on the right has been styled with CSS.



Password field

A ***password field*** is a widget for input elements with the `type` attribute of "password", which allows users to enter a password without the password contents being displayed on-screen. Web browsers usually provide facilities to remember passwords at various websites to help users.

Forms that submit passwords or any sensitive data should always submit with URLs that use HTTPS. Form data submitted with HTTP are not encrypted, but HTTPS encrypts form data.

The HTML below uses the `size` attribute to limit the password field's width and uses the `maxlength` attribute to limit the maximum number of characters the user can enter. The `size` and `maxlength` attributes can be used on text boxes as well.

Figure 3.3.4: Password field that limits the number of characters to 10.

```
<input type="password" name="secret" size="10"
maxlength="10">
```

Feedback?

Fieldset

The ***<fieldset>*** tag groups related form widgets together and draws a box around the related widgets. The ***<legend>*** tag defines a caption for a `<fieldset>`.

Figure 3.3.5: Fieldset around related radio buttons.

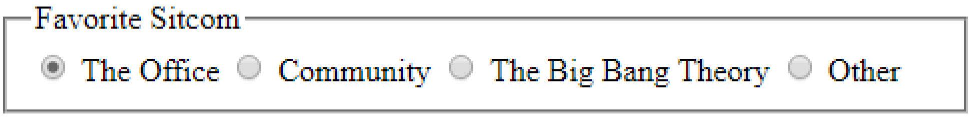
```
<fieldset>
  <legend>Favorite Sitcom</legend>

  <input type="radio" name="sitcom" value="The Office"
id="theOffice">
  <label for="theOffice">The Office</label>

  <input type="radio" name="sitcom" value="Community" id="community">
  <label for="community">Community</label>

  <input type="radio" name="sitcom" value="The Big Bang Theory"
id="bigBang">
  <label for="bigBang">The Big Bang Theory</label>

  <input type="radio" name="sitcom" value="Other" id="other">
  <label for="other">Other</label>
</fieldset>
```



Favorite Sitcom

● The Office ○ Community ○ The Big Bang Theory ○ Other

Feedback?

PARTICIPATION ACTIVITY | 3.3.5: Menus, buttons, and passwords. ✓

1) A `<fieldset>` can group drop-down menus, buttons, and other widgets.

● True

○ False

✓ **Correct**

Any type of widget can be grouped in a `<fieldset>`. Fieldsets and legends benefit individuals that use assistive technologies like screen readers. CSS can be used to remove the rectangle if desired.

2) A drop-down menu only allows one option to be selected.

● True

○ False

✓ **Correct**

If the `multiple` attribute is set in the `<select>` tag, the drop-down menu changes into a list box. Only a list box allows users to select multiple options.

3) An `<option>` tag must have a `value` attribute.

○ True

● False

✓ **Correct**

If the `<option>` tag does not have a value, the browser will send the option text as the value.

4) Which element can create a button with an image?

● <button>

○ <input>

Correct

The `<button>` element gives developers more control over the button's appearance than the `<input>` element.

✓

5) Buttons always submit form data to a server.

○ True

● False

Correct

While a submit button causes form data to be sent to the server, buttons can be used for other purposes, such as running JavaScript code.

✓

6) Passwords from the password widget that are sent to the server using HTTP are safe from prying eyes.

○ True

● False

Correct

The browser doesn't display the password on the screen, but HTTP sends the password over the network as text, which is viewable by anyone watching the network traffic between the web browser and the server. To keep passwords secure, the form must use HTTPS.

✓

7) Which input element attribute limits the number of characters the user can type in a text box or password field?

○ size

● maxlength

Correct

The `size` attribute limits the width of the text box, but the `maxlength` attribute limits the number of characters that can appear in a text box.

✓

Feedback?

CHALLENGE ACTIVITY | 3.3.1: Building common widgets. ✓

530096.4000608.qx3zqy7

Jump to level 1

✓ 1

✓ 2

✓ 3

✓ 4

Add a checkbox associated with each label. The first checkbox should have name and id of "likeCats" and be initially not checked. The second checkbox should have name and id of "likeDogs" and be initially checked. **SHOW EXPECTED**

```
<form action="https://wp.zybooks.com/form-viewer.php" target=

   <p>
      <label for="likeCats">Like cats?</label>
      <input type = "checkbox" name = "likeCats" id = "likeCa
      <!-- TODO: Add checkbox -->
   </p>
```

```

  <p>
     <label for="likeDogs">Like dogs?</label>
     <input type = "checkbox" name = "likeDogs" id = "likeDo
     <!-- TODO: Add checkbox -->
  </p>


  <p><input type="submit" value="Submit"></p>
```

1 2 3 4

Check Next

✓

✓ Testing number of <input> with type of "checkbox"

Yours 2

✓ Testing name attribute of first checkbox

Yours `likeCats`

✓ Testing id attribute of first checkbox

Yours `likeCats`

✓ Testing whether first checkbox is checked

Yours `Is not checked`

✓ Testing name attribute of second checkbox

Yours `likeDogs`

✓ Testing id attribute of second checkbox

Yours `likeDogs`

✓ Testing whether second checkbox is checked

Yours `Is checked`

Your webpage

Like cats? □

Like dogs? ☑

Submit

View your last submission ^

```
 <p>
   <label for="first">Like cats?</label>
   <input type="checkbox" name="first" id="first" checked>
</p>

<p>
   <label for="second">Like dogs?</label>
   <input type="checkbox" name="second" id="second">
</p>
```

Feedback?

Exploring further:

- HTML <input> tag from W3Schools
- HTML <select> tag from W3Schools
- HTML <button> tag from W3Schools

How was this section? 👍 👎 **Provide section feedback**