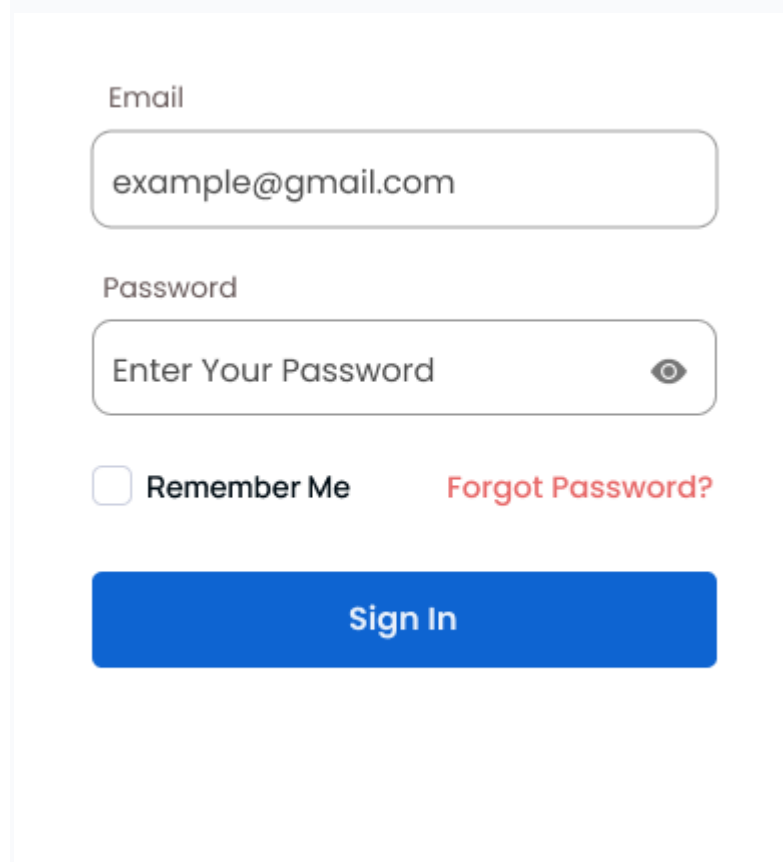


# HTML Form

An HTML Form is a section of the document that collects input from the user. The input from the user is generally sent to a server (Web servers, Mail clients, etc). We use the HTML `<form>` element to create forms in HTML.



The image shows a sample HTML form. It has a white background with rounded corners. At the top, there is a label "Email" above a text input field containing "example@gmail.com". Below that is a label "Password" above a password input field containing "Enter Your Password" and a toggle icon (an eye). Under the password field, there is a checkbox labeled "Remember Me" and a link "Forgot Password?" in red text. At the bottom, there is a blue button with the text "Sign In".

## Example: HTML Form

The HTML `<form>` element is used to create HTML forms. For example,

```
<form>
```

```
  <label for="firstname">First name: </label>
```

```
  <input type="text" name="firstname" required>
```

```
<br>
```

```
<label for="lastname">Last name: </label>
```

```
<input type="text" name="lastname" required>
```

```
<br>
```

```
<label for="email">email: </label>
```

```
<input type="email" name="email" required>
```

```
<br>
```

```
<label for="password">password: </label>
```

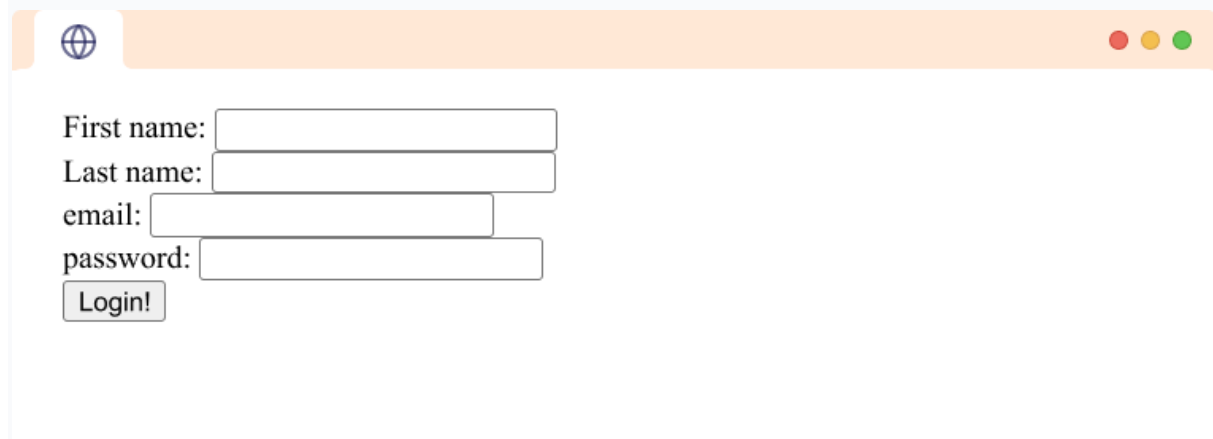
```
<input type="password" name="password" required>
```

```
<br>
```

```
<input type="submit" value="Login!">
```

```
</form>
```

## Browser Output

A screenshot of a web browser window with a light orange header bar. The browser's address bar shows a globe icon. The main content area displays a login form with the following elements: a label 'First name:' followed by a text input field; a label 'Last name:' followed by a text input field; a label 'email:' followed by a text input field; a label 'password:' followed by a password input field; and a 'Login!' button at the bottom.

## HTML Form Elements

A form contains special interactive elements that users use to send the input. They are text inputs, textarea fields, checkboxes, dropdowns, and much more. For example,

```
<form>
```

```
<label for="name">Name :</label>
```

```
<input type="text" name="name"><br><br>
```

```
<label for="sex">Sex:</label>
```

```
<input type="radio" name="sex" id="male" value="male">
<label for="male">Male</label>
<input type="radio" name="sex" id="female" value="female">
<label for="female">Female</label> <br><br>
<label for="country">Country: </label>
<select name="country" id="country">
  <option>Select an option</option>
  <option value="nepal">Nepal</option>
  <option value="usa">USA</option>
  <option value="australia">Australia</option>
</select><br><br>
<label for="message">Message:</label><br>
<textarea name="message" id="message" cols="30"
rows="4"></textarea><br><br>
<input type="checkbox" name="newsletter" id="newsletter">
<label for="newsletter">Subscribe?</label><br><br>
<input type="submit" value="Submit">
</form>
```

Browser Output

A screenshot of a web browser window with a light orange header bar. The browser's address bar shows a globe icon. The form contains the following elements: a 'Name:' label followed by a text input field; a 'Sex:' label followed by two radio buttons labeled 'Male' and 'Female'; a 'Country:' label followed by a dropdown menu showing 'Select an option'; a 'Message:' label followed by a large text area; a checkbox labeled 'Subscribe?'; and a 'Submit' button.

To learn more about the various form controls, visit [HTML Form Inputs](#).

## Form Attributes

The HTML `<form>` element contains several attributes for controlling data submission. They are as follows:

### action

The `action` attributes define the action to be performed when the form is submitted. It is usually the url for the server where the form data is to be sent.

```
<form action="/login">
```

```
  <label for="email">Email:</label>
```

```
  <input type="email" name="email"><br><br>
```

```
  <label for="password">Password:</label>
```

```
<input type="password" name="password"><br><br>
```

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

```
</form>
```

In the above example, when the form is submitted, the data from the form is sent to `/login`.

method

The `method` attribute defines the HTTP method to be used when the form is submitted. There are 3 possible values for the `method` attribute:

**post** - It is used to send data to a server to update a resource.

```
<form method = "post">
```

```
...
```

- ```
</form>
```

**get**: It is used to request data from a specified resource.

```
<form method = "get">
```

```
...
```

- ```
</form>
```
- **dialog**: This method is used when the form is inside a `<dialog>` element. Using this method closes the dialog and sends a form-submit event.

To learn more about HTTP methods GET and POST, visit *HTML Form Action: POST and GET*.

target

It specifies where to display the response received after the form is submitted. Similar to the `target` attribute in `<a>` tags, the `target` attribute has four possible values.

`_self` (default): Load the response into the same browser tab.

```
<form target="_self">
```

```
  <label for="firstname">Enter your first name:</label>
```

```
  <input type="text" name="firstname">
```

- ```
</form>
```

`_blank`: Load the response into a new browser tab.

```
<form target="_blank">
```

```
  <label for="firstname">Enter your first name:</label>
```

```
  <input type="text" name="firstname">
```

- ```
</form>
```

`_parent`: Load into the parent frame of the current one. If no parent is available, it loads the response into the same tab.

```
<form target="_parent">
```

```
  <label for="firstname">Enter your first name:</label>
```

```
  <input type="text" name="firstname">
```

- ```
</form>
```

`_top`: Load the response into the top-level frame. If no parent is available, it loads the response into the same tab.

```
<form target="_top">
```

```
  <label for="firstname">Enter your first name:</label>
```

```
  <input type="text" name="firstname">
```

- ```
</form>
```

enctype

It specifies how the form data should be encoded for the request. It is only applicable if we use the POST method.

```
<form method="post" enctype="application/x-www-form-urlencoded">
</form>
```

In the above example, data from the form will be encoded in the x-www-form-urlencoded format (which is the default encoding format).

name

It specifies the name of the form. The `name` is used in Javascript to reference or access this form.

```
<form name="login_form">
  <label for="email">Email:</label>
  <input type="email" name="email"><br><br>
  <label for="password">Password:</label>
  <input type="password" name="password"><br><br>
  <input type="submit" value="Submit">
</form>
```

The above form can be accessed in javascript as:

```
document.forms['login_form']
```

Although it is possible to use name to access form elements in javascript, it is recommended to use `id` to access the form elements.

novalidate

If the `novalidate` attribute is set, all validations in the form elements are skipped.

```
<form novalidate>
```

```
<label for="email">Enter your email:</label>
```

```
<input type="email" id="email" name="email"><br><br>
```

```
<input type="submit">
```

```
</form>
```

In the above example, the form will be submitted even if we enter some invalid value to the `email` field, such as `Hi`.