**Submit**

You have recently learned about how forms are sent to web servers and the difference between Get and Post. In this reading, you will build on this knowledge by learning about Submit.

**Action and method**

Form submissions are an essential part of the world wide web. Nearly every website uses forms, from buying items online to ordering food for delivery. When you click the login button on a website, it sends your username and password to a web server to log you into your account. As you know by now, you add a form to your web page using the form tag.

1

2





But how the form is submitted is determined by two essential attributes: action and method. The action attribute specifies to which web address the form must be sent. This is address is location of server-side code that will process the request.

1

2





It is important to note that action can be a full URL address such as *https://meta.com*, an absolute path such as */login*, or a relative path such as *login*. The absolute path, which starts with a forward slash, will use the base address of the current website, such as *https://meta.com* and combine it with the absolute path. For example, if */login* is the absolute path, the form will be submitted to *https://meta.com/login*. If the address is *https://meta.com/company-info/* and */login* is the absolute path, the submission address will still be *https://meta.com/login*. Similarly, a relative path will combine the current web address with a relative path. For example, if the web browser is currently on the web page *https://meta.com/company-info/*, and the relative path is set to *login*, the form will be submitted to *https://meta.com/company-info/login*. The method attribute specifies which HTTP method is used to submit the form; GET or POST.

1

2





1

2





The form will default to the HTTP GET method when the method attribute is not provided. As you may already know, when the form is submitted using the HTTP GET method, the data in the form's fields are encoded in the URL. And when the form is submitted using the HTTP POST method, the data is sent as part of the HTTP request body. When the web server receives the request, it processes the data and sends back an HTTP response. The response indicates the result of the submission, which can be successful or fail due to invalid or incorrect data. However, as a front-end developer, it is essential to know that forms are not the only way to submit data to the web server. As you learn more about JavaScript and front-end libraries, you’ll discover that developers often submit HTTP requests directly via code and send data as part of the HTTP request body in a text format called JavaScript Object Notation, or JSON. But that is a topic for another course. For now, practice building HTML forms and submitting data to the web server using the different attributes available.

<form>

</form>

<form action="/login">

</form>

<form method="get">

</form>

<form method="post">

</form>