pushState() takes three parameters: a state object, a title (which is currently ignored), and (optionally) a URL. Let's examine each of these three parameters in more detail:

* **state object** — The state object is a JavaScript object which is associated with the new history entry created by pushState(). Whenever the user navigates to the new state, a popstate event is fired, and the state property of the event contains a copy of the history entry's state object.

The state object can be anything that can be serialized. Because Firefox saves state objects to the user's disk so they can be restored after the user restarts the browser, we impose a size limit of 640k characters on the serialized representation of a state object. If you pass a state object whose serialized representation is larger than this to pushState(), the method will throw an exception. If you need more space than this, you're encouraged to use sessionStorage and/or localStorage.

* **title** — Firefox currently ignores this parameter, although it may use it in the future. Passing the empty string here should be safe against future changes to the method. Alternatively, you could pass a short title for the state to which you're moving.
* **URL** — The new history entry's URL is given by this parameter. Note that the browser won't attempt to load this URL after a call to pushState(), but it might attempt to load the URL later, for instance after the user restarts the browser. The new URL does not need to be absolute; if it's relative, it's resolved relative to the current URL. The new URL must be of the same origin as the current URL; otherwise, pushState() will throw an exception. This parameter is optional; if it isn't specified, it's set to the document's current URL.