Learn jQuery: Event Handlers

jquery on event listeners

jQuery `.on()` event listeners support all common browser *event types* such as mouse events, keyboard events, scroll events and more.

```
// A mouse event 'click'
$('#menu-button').on('click', () => {
  $('#menu').show();
});

// A keyboard event 'keyup'
$('#textbox').on('keyup', () => {
  $('#menu').show();
});

// A scroll event 'scroll'
$('#menu-button').on('scroll', () => {
  $('#menu').show();
});
```

jquery event object

In a jQuery event listener, an *event object* is passed into the event handler callback when an event occurs. The *event object* has several important properties such as `type` (the event type) and `currentTarget` (the individual DOM element upon which the event occurred).

```
// Hides the '#menu' element when it has been clicked.
$('#menu').on('click', event => {
  // $(event.currentTarget) refers to the '#menu' element that was clicked.
  $(event.currentTarget).hide();
});
```

jquery event.currentTarget attribute

In a jQuery event listener callback, the `event.currentTarget` attribute only affects the individual element upon which the event was triggered, even if the listener is registered to a group of elements sharing a class or tag name.

```
// Assuming there are several elements with the
// class 'blue-button', 'event.currentTarget' will
// refer to the individual element that was clicked.
$('.blue-button').on('click', event => {
  $(event.currentTarget).hide();
});
```

jquery on method chaining

jQuery `.on()` event listener methods can be chained together to add multiple events to the same element.

```
// Two .on() methods for 'mouseenter' and
// 'mouseleave' events chained onto the
// '#menu-button' element.
$('#menu-button').on('mouseenter', () => {
  $('#menu').show();
}).on('mouseleave', () => {
  $('#menu').hide();
});
```

jquery on method

The jQuery `.on()` method adds *event listeners* to jQuery objects and takes two arguments: a string declaring the event to listen for (such as 'click'), and the event handler callback function. The `.on()` method creates an event listener that detects when an event happens (for example: when a user clicks a button), and then calls the *event handler* callback function, which will execute any defined actions after the event has happened.

```
//The .on() method adds a 'click' event to the '#login' element. When the '#login' element is clicked, the callback function will be executed, which reveals the '#login-form' to the user.

$('#login').on('click', () => {
  $('#login-form').show();
});
```

Print Share ▼