

# this in jQuery

L'operatore `this` in JavaScript si riferisce all'oggetto corrente. Poichè gli elementi HTML vengono considerati come oggetti (o più tecnicamente nodi oggetto), questo operatore viene usato anche per gli elementi della pagina. jQuery utilizza la stessa sintassi, ma con alcune differenze. Vediamo quali.

Partiamo da un esempio JavaScript che sicuramente avrete già avuto modo di osservare su molti siti:

```
const button = document.getElementById('btn');

button.onclick = () => {

    this.className = 'pressed';

};
```

In questo caso `this` fa riferimento a `button`, e quindi la proprietà `className` può essere usata utilizzando direttamente `this`.

Ora riscriviamo l'esempio con jQuery:

```
var $button = $( "#btn" );

$button.click(function() {

    $( this ).addClass( "pressed" );

});
```

`this` viene racchiuso all'interno del wrapper `$( )` e per questo motivo non fa più riferimento ad un nodo oggetto (come nel primo esempio), ma è divenuto un riferimento ad un oggetto jQuery.

In questo caso la proprietà `className` non è più accessibile perchè non fa parte delle proprietà di un oggetto jQuery. Ma se facciamo un passo indietro e non usiamo il wrapper di jQuery, la proprietà è di nuovo accessibile:

```
var $button = $( "#btn" );

$button.click(function() {

    this.className = "pressed";

});
```

Un metodo alternativo è quello di usare la sintassi `$( elemento )[0]` che trasforma un oggetto jQuery in un normale nodo oggetto:

```
var $button = $( "#btn" );

$button.click(function() {

    $( this )[0].className = "pressed";

});
```

Questo risultato si può anche ottenere utilizzando la sintassi `$( elemento ).get( 0 )`, o se sono più elementi `$( elementi ).get()`.