

GABRIELE ROMANATO

jQuery: differenze nel calcolo della posizione degli elementi tra i metodi `position()` e `offset()`

La differenza tra i metodi `position()` e `offset()` di jQuery sta nel modo con cui vengono calcolate le coordinate restituite.

Nel primo caso, le coordinate vengono calcolate sempre rispetto al genitore dell'elemento, mentre nel secondo caso tali coordinate possono essere calcolate rispetto all'offset dell'elemento genitore rispetto alla pagina quando il genitore è posizionato, ossia ha un valore CSS diverso da `static`. Se il genitore non è posizionato, entrambi i metodi resituiscono gli stessi valori. Vediamo un esempio.

Abbiamo la seguente struttura:

```
<div id="container">
    <div id="test"></div>
</div>
```

e i seguenti stili:

```
#container {
    width: 600px;
    height: 400px;
    background: #ccc;
    position: relative;
}

#test {
    width: 100px;
    height: 100px;
    background: #d34545;
```

```
        position: relative;
    }
}
```

In questo caso il genitore è posizionato. Se proviamo ad aggiungere 50 pixel ai valori delle proprietà `top` e `left` restituite da questi metodi, noteremo sicuramente la differenza:

```
$(function() {

    var test = $( "#test", "#container" );
    var triggerPosition = $( "#position" );
    var triggerOffset = $( "#offset" );

    triggerPosition.click(function() {

        test.css({ top: 0, left: 0 });

        var top = test.position().top;
        var left = test.position().left;

        test.animate({
            top: top + 50,
            left: left + 50
        }, "slow" );

        return false;

    });

    triggerOffset.click(function() {

        test.css({ top: 0, left: 0 });

        var top = test.offset().top;
        var left = test.offset().left;

        test.animate({
            top: top + 50,
            left: left + 50
        }, "slow" );

    });

});
```

```
        return false;  
  
    });  
  
});
```

Con il primo metodo otteniamo una distanza inferiore rispetto al secondo, pur avendo specificato gli stessi valori da aggiungere per entrambi. Questo accade perchè nel caso di `offset()` le coordinate sono il risultato della somma dell'offset del genitore rispetto alla pagina e del nostro valore.