

DRAG & DROP WITH SNAP OPTION

http://www.tutorialspoint.com/script.aculo.us/scriptaculous_snap_example.htm

Copyright © tutorialspoint.com

Description

This option is used to cause a draggable to snap to a grid or to constrain its movement.

- If it is set to false *default*, no snapping or constraining occurs.
- If an integer *x*, the draggable will snap to a grid of *x* pixels.
- If an array [*x*, *y*], the horizontal dragging will snap to a grid of *x* pixels and the vertical will snap to *y* pixels.
- It can also be a function conforming to Function*x, y, draggable* that returns an array [*x*, *y*].

Syntax

Here are various syntaxes to use the *snap* option.

```
// Snap target to a 50-pixel grid while dragging
new Draggable('element', {snap:50});

OR

// Constrain dragging to a 100x50px box
new Draggable('element', {
  snap: function(x, y) {
    return[ (x < 100) ? (x > 0 ? x : 0) : 100, (y < 50) ? (y > 0 ? y : 0) : 50 ];
  }
});

OR

// Constrain dragging to element's parent node
new Draggable('element', {
  snap: function(x, y, draggable) {
    function constrain(n, lower, upper) {

      if (n > upper)
        return upper;
      else if (n < lower)
        return lower;
      else
        return n;
    }

    var element = draggable.element.getDimensions( );
    var parent = draggable.element.parentNode.getDimensions( );
    return [
      constrain(x, 0, parent.width - element.width),
      constrain(y, 0, parent.height - element.height)
    ];
  }
});
```

Example

```
<html>
  <head>
    <title>Draggables Elements</title>

    <script type="text/javascript" src="/javascript/prototype.js"></script>
    <script type="text/javascript" src="/javascript/scriptaculous.js"></script>
```

```

<script type="text/javascript">
    window.onload = function() {
        new Draggable('myimage',
            {
                revert:true, snap: function(x, y) {
                    return[ (x < 100) ? (x > 0 ? x : 0 ) : 100, (y < 50) ? (y > 0 ? y :
0) : 50 ];
                }
            }
        );
    }
</script>

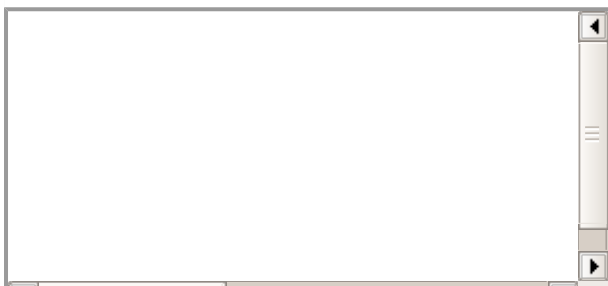
</head>
<body>

<p>Try to drag the following image out of its defined
    boundary and see the result. Later change its boundary and
    repeat the exercise.</p>

<img />
</body>
</html>

```

This will produce following result –



Loading [MathJax]/jax/output/HTML-CSS/fonts/TeX/fontdata.js