

Drag & Drop

Introduction

- The Drag & Drop API allows us to develop web sites in which the user will be able to drag and drop items on screen.
- The Drag & Drop API allows the user to drag and drop the items either to a new location within the same web browser window or to a new location within another window on the screen, even of another running application, such as a text editor or a graphics design application.

Introduction

- The links, text nodes and images are all draggable by default.
- We can define a drop zone by specifying the `onDragOver` and the `onDrop` events handlers.
- The `onDragOver` event handler function will be invoked when the mouse is over it during a dragging operation. If we don't want to allow dragging over then the function we assign this event handler should return `false`.

Introduction

- The `onDrop` event handler specifies what should be done with the dropped data.

Browser Support

- Each web browser supports the Drag & Drop capability in a different way.
- The Safari web browser is the simplest to handle.

Sample

```
<title>Simple drag demo</title>
<style>
#rectangle { height: 200px; border: 5px solid #ccc; }
</style>

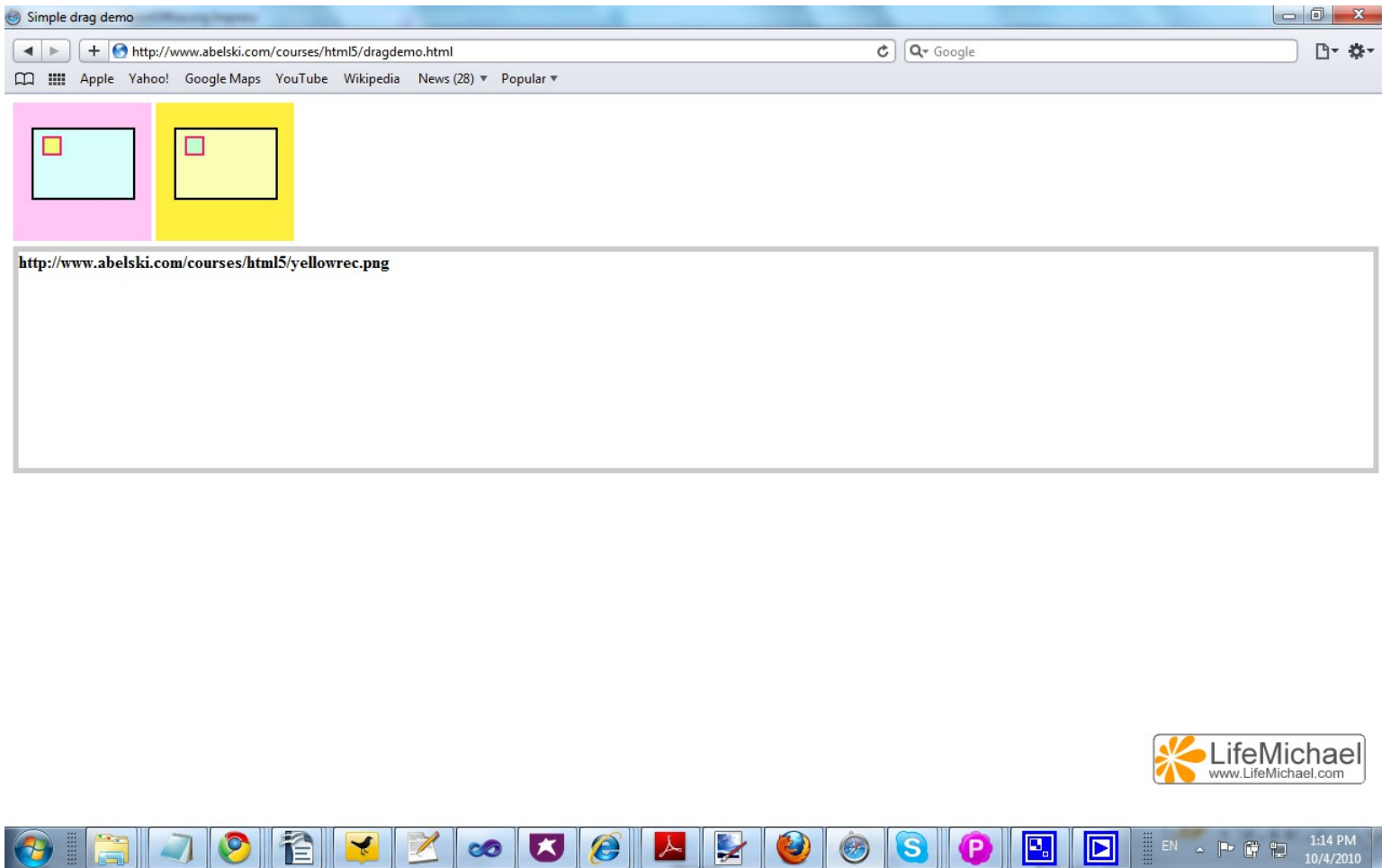

<div id="rectangle"></div>
<script>
var ob = document.getElementById('rectangle');
ob.innerHTML = "<b>...</b>";
ob.ondrop = function (event)
{
    this.innerHTML = '<b>'+event.dataTransfer.getData('Text')+'</b>';
};
ob.ondragover = function ()
{
    return false;
};
</script>
```

[Click Here](#)



This Demo Works on Safari Web Browser Only!

Sample



Drag & Drop

06/24/14

© 2008 Haim Michael. All Rights Reserved.

1

Introduction

- The Drag & Drop API allows us to develop web sites in which the user will be able to drag and drop items on screen.
- The Drag & Drop API allows the user to drag and drop the items either to a new location within the same web browser window or to a new location within another window on the screen, even of another running application, such as a text editor or a graphics design application.

Introduction

- The links, text nodes and images are all draggable by default.
- We can define a drop zone by specifying the `onDragOver` and the `onDrop` events handlers.
- The `onDragOver` event handler function will be invoked when the mouse is over it during a dragging operation. If we don't want to allow dragging over then the function we assign this event handler should return false.

Introduction

- The `onDrop` event handler specifies what should be done with the dropped data.

Browser Support

- Each web browser supports the Drag & Drop capability in a different way.
- The Safari web browser is the simplest to handle.

Sample

```
<title>Simple drag demo</title>
<style>
#rectangle { height: 200px; border: 5px solid #ccc; }
</style>


<div id="rectangle"></div>
<script>
var ob = document.getElementById('rectangle');
ob.innerHTML = "<b>...</b>";
ob.ondrop = function (event)
{
    this.innerHTML = '<b>'+event.dataTransfer.getData('Text')+'</b>';
};
ob.ondragover = function ()
{
    return false;
};
</script>
```

[Click Here](#)



This Demo Works on Safari Web Browser Only!

Sample

