

HTML5 拖放

拖放 (Drag & Drop)

拖放是一種常見的特性，即抓取對象以後，拖到另一個位置放下。
在 HTML5 中，拖放是標準的一部分，任何元素都能夠拖放。

Html5 拖放實例

下面的例子是一個簡單的拖放實例：

```
<!DOCTYPE HTML>

<html>

<head>

<script type="text/javascript">

function allowDrop(ev)

{

ev.preventDefault();

}

function drag(ev)

{

ev.dataTransfer.setData("Text",ev.target.id);

}

function drop(ev)

{

ev.preventDefault();

var data=ev.dataTransfer.getData("Text");

ev.target.appendChild(document.getElementById(data));

}
```

```
</script>

</head>

<body>

<div id="div1" ondrop="drop(event)"

ondragover="allowDrop(event)"></div>



</body>

</html>
```

我們將上述的例子，依照不同的拖放事件作分別敘述。

設置元素可拖放

首先，為了使元素可拖動，把 `draggable` 屬性設為 `true`：

```
<img draggable="true" />
```

拖動什麼 - `ondragstart` 和 `setData()`

然後，規定當元素被拖動時，會發生什麼。

在上面的例子中，`ondragstart` 屬性調用了一個函數，`drag(event)`，它規定了被拖動的數據。

`dataTransfer.setData()` 方法設置被拖數據的類型和值：

```
function drag(ev)

{

ev.dataTransfer.setData("Text",ev.target.id);

}
```

在這個例子中，數據類型是 `"Text"`，值是可拖動元素的 `id` (`"drag1"`)。

放到何處 - ondragover

ondragover 事件規定在何處放置被拖動的數據。

系統初始設定無法將數據/元素放置在其他元素中。如果需要設置允許放置，我們必須阻止對元素的默認處理方式。

這要通過調用 ondragover 事件的 event.preventDefault() 方法

```
event.preventDefault()
```

進行放置 - ondrop

當放置被拖數據時，會發生 drop 事件。

在上面的例子中，ondrop 屬性調用了一個函數，drop(event)：

```
function drop(ev)

{

ev.preventDefault();

var data=ev.dataTransfer.getData("Text");

ev.target.appendChild(document.getElementById(data));

}
```

代碼解釋

- 調用 preventDefault() 來避免瀏覽器對數據的默認處理(drop 事件的默認行為是以連結形式打開)
- 通過 dataTransfer.getData("Text") 方法獲得被拖的數據，該方法將返回在 setData() 方法中設置為相同類型的任何數據。
- 被拖數據是被拖元素 id("drag1")
- 把被拖元素追加到放置元素(目標元素)中

資料來源：

<http://www.w3school.com.cn/>