

HTML5 新特性 -- Unit04

1. CanvasRenderingContext2D 接口

1.1 measureText 属性

`measureText` 属性将返回一个可测量的文本对象，其语法结构是：

```
TextMetrics CanvasRenderingContext2D.measureText(string text)
```

1.2 TextMetrics 对象

1.2.1 width 属性

`width` 属性用于获取文本的宽度，语法结构是：

```
variable = TextMetrics.width
```

2. 拖放

拖放是 HTML5 的新特性，即将对象从一个位置抓取到另外的一个位置，在 HTML5 任何元素都可以拖放。

为解决浏览器的兼容性，建议在被拖放的对象上添加 `draggable="true"` 属性。

2.1 DragEvent 事件接口

`DragEvent` 是表示拖、放事件的接口，该接口继承自 `MouseEvent` 和 `Event` 接口。

`DragEvent` 接口的事件包括：

2.1.1 源对象事件

- `dragstart`

`dragstart` 事件在源对象开始被拖动时触发，其语法结构是：

```
HTMLElement.addEventListener('dragstart', function(){  
    ...  
})
```

该事件在拖放过程中只能被触发一次

示例代码如下：

```
//获取段落对象
var pEle = document.getElementById('p');
//开始拖放的事件
pEle.addEventListener('dragstart',function(){
    console.log('开始拖动段落了');
});
```

• drag

drag 事件在源对象被拖放过程中触发，其语法结构是：

```
HTMLElement.addEventListener('drag',function(){
    ...
})
```

该事件在拖放过程中反复被触发

示例代码如下：

```
//获取段落对象
var pEle = document.getElementById('p');
//拖放过程中
pEle.addEventListener('drag',function(){
    console.log('你把我拽到哪儿?');
});
```

• dragend

dragend 事件在源对象被释放时触发（可能在目标区域内，也可能在目标区域外），其语法结构是：

```
HTMLElement.addEventListener('dragend',function(){
    ...
})
```

该事件只能被触发一次

2.1.2 目标对象事件

• dragenter

dragenter 事件在源对象进入目标对象时触发，其语法结构是：

```
HTMLElement.addEventListener('dragenter',function(){
    ...
})
```

该事件可能会反复执行多次

• dragover

dragover 事件在源对象在目标对象上悬停时触发，其语法结构是：

```
HTMLElement.addEventListener('dragover',function(event){
    event.preventDefault();
    ...
})
```

在该事件中必须调用事件对象的 `preventDefault()` 方法，否则不会触发 drop 事件

• dragleave

dragleave 事件在源对象拖动离开目标对象时触发，其语法结构是：

```
HTMLElement.addEventListener('dragleave',function(){
    ...
})
```

该事件可能会反复执行多次

• drop

drop 事件在源对象在目标对象上释放时触发，其语法结构是：

```
HTMLElement.addEventListener('drop',function(){
    ...
})
```

该事件只能被触发一次

在 drop 事件中也"必须"要执行 `event.preventDefault()` 及 `event.stopPropagation()` 方法，否则源对象的行为可能会触发（如拖动目标为图像的话，在 Firefox 浏览器中会在新标签页打开）

示例代码如下：

```
targetEle.addEventListener('drop',function(event){
    event.preventDefault();
    event.stopPropagation();
});
```

以上事件按正常触发顺序是：

`dragstart`、`drag`、`dragenter`、`dragover`、`drop`、`dragend`

2.3 DataTransfer 对象

`DragEvent` 接口的 `dataTransfer` 属性将返回 `DataTransfer` 对象，该对象用于保存拖放数据，其语法结构是：

```
DataTransfer DragEvent.dataTransfer
```

• setData() 方法

`setData()` 方法用于为一个给定的类型设置数据，如果数据不存在，则自动添加到末尾，如果存在，则替换原始数据，其语法结构是：

```
DataTransfer.setData(type,value)
```

所谓的类型其实就是一个键名

• getData() 方法

`getData()` 方法用于返回指定类型的数据，如果类型不存在或者没有数据，则返回空字符串，其语法结构是：

```
variable = DataTransfer.getData(type)
```

作业：将自定义视频播放器与弹幕组合