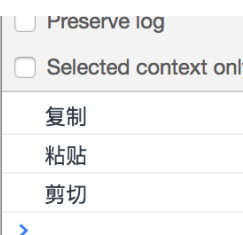


# API 介绍

浏览器的三个事件

- 复制 oncopy
- 粘贴 onpaste
- 剪切 oncut

```
document.body.oncut = e => {  
  console.log("剪切")  
}  
document.body.oncopy = e => {  
  console.log("复制")  
}  
document.body.onpaste = e => {  
  console.log("粘贴")  
}
```



三个事件可作用在body上也可以监听DOM。



clipboardData对象:用于访问以及修改剪贴板中的数据。

兼容ie:

```
let data = (e.clipboardData || window.clipboardData);
```

使用：

对象有三个方法: `getData()`、`setData()`、`clearData()`

`getData()`：访问剪切板里的数据，只接受一个参数，  
`data.getData("text")`。配合粘贴事件`onpaste`使用。打印要粘贴的数据。

复制/剪切时获取数据：`window.getSelection(0).toString()`；只能在出发复制`oncopy`和剪切`oncut`时触发。

`setData()`：设置剪切板里的数据，第一个参数也是 'text'，第二个参数是要替换上的文本

## 应用：

原理很简单，在浏览器复制文本时`oncopy`时取消默认事件，在用`clipboardData`对象的`setData`替换掉文本即可。

```
document.body.oncopy = event => {
  event.preventDefault();
  let textFont, copyFont = window.getSelection(0).toString();
  if (copyFont.length > 5) {
    textFont = copyFont + "\n转载请注明出处"
  } else {
    textFont = copyFont;
  }
  if (event.clipboardData) {
    return event.clipboardData.setData('text', textFont);
  } else {
    return window.clipboardData.setData("text", textFont);
  }
}
```

## 浏览器禁用右键菜单，复制粘贴

# 剪切，文本选择等功能。

右键菜单禁用

```
document.body.oncontextmenu =function (e){  
    e.preventDefault()  
    //return false  
}
```

文本选择禁用

```
document.body.onselectstart =function (e){  
    e.preventDefault()  
    //return false  
}
```

// css 禁止文本选择 这样不会触发js

```
body {  
    user-select: none;  
    -moz-user-select: none;  
    -webkit-user-select: none;  
    -ms-user-select: none;  
}
```

直接使用return false 可以省去对e的访问。不过在剪切板中添加版权信息，和右键菜单自定义中还是要使用e.preventDefault()来禁用掉默认事件。

