json 异步加载 时间线

1. 异步加载方式defer和async（一般为同步加载，掉包整个页面阻塞，解决阻塞问题）：

（1）<script defer=’defer’></script> //IE加载方式,允许script标签中写js代码

（2）<script async=’async’></script> //其他浏览器加 载方式，不允许script标签中写代码

2、错误用法：

（1）同样一个文件不同命令加载

<head>

<script defer=’defer’src=’tool.js’></script>

<script async=’async’ src=’tool.js’></script>

</head>

原因：三个script标签都是异步加载，并分开执行，可能导致内部代码混乱，一次执行变多次执行状态，最后结果无法预料

（2） 加载时需要发送接收请求，所以加载完成后，后面的程序已经执行导致的矛盾

var script=document.createElement(‘script’);  
script.type=’text/javascript’;

script.src=’tool.js’; //里面有test函数

test(); //tool.js还没下载完，test没加载就执行，会报错

1. 常用异步加载方式：DOM操作，解决与后续代码矛盾（IE script标签没有onload方法）

function asyncLoad(){  
 var script=document.createElement(‘script’);  
 script.type=’text/javascript’;

script.src=’tool.js’;

if(script.readystate){//状态码  
 script.onreadystatechange=function(){

if(script.readystate==’loaded’||script.readystate==’complete’){test();}

else{

script.onload=function(){test();//tool.js标签里的方法}

}}  
}