今天学习的内容

1:复习昨天知识重点

AJAX处理JSON 格式的数据

JSON-->原生AJAX

php 处理 json

header("Content-Type:application/json;charset=utf-8");

$input = json\_encode($rows);echo $input;

js 处理 json var obj = JSON.parse(xhr.responseText)

JQUERY中AJAX方法

$.ajax({

type:"GET",

url:"x.php",

data:{ename:"tom",age:19},

success:function(data){ //200 OK 自动JSON.parse();

},

error:function(){//404 500

}

});

2:jquery中ajax其它方法与跨域访问

**jquery中ajax其它方法**

$.get(url,data,fn); 发送get请求

fn:回调函数{发送请求并且接收返回数据后调用方法}

$.post(url,data,fn); 发送post请求

$("选择器").load(url,data,fn); \*\*\*

$.getScript(url,data,fn); 发送get请求，接收响应数据格式为script,会自动执行 eval(xhr.responseText)

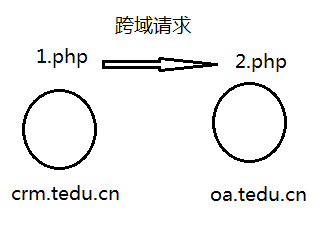
$.getJSON(url,data,fn); 发送get请求，接收响应数据格式为json,会自动执行JSON.parse();

$("选择器").load(url); 发送get请求，接收响应数据格式必须为HTML,会动将html保存选择器的标签中间.

2.2跨域访问

Cross Domain Request

从一个域名下的某个资源访问另一个域名下的某个资源



(1):两个域名不同 crm.tedu.cn--->oa.tedu.cn

(2):协议名不同 http://tedu.cn/1.php--> https://tedu.cn/2.php

(3):主机名/IP不同 http://localhost/1.php-->http://127.0.0.1/2.php

(4):端口号不同

http://127.0.0.1:8080/1.php->http://127.0.0.1:80/2.php

浏览器允许跨域请求情形

img/link/script/iframe 等标签可以

浏览器不允许跨域请求情形

浏览器默认禁止XHR跨域请求-跨域访问可能导致外来的数据对当前程序影响，造成安全风险.

解决方案

JSONP JSON with Padding 填充式JSON，是一种使JSON数据的方法，用于解决浏览器的XHR跨域请求限制.

JSONP 的思路

发起异步请求不使用XHR，使用一个动态创建script标签代替XHR，<script async src="跨域访问x.php"></script>要求服务器端必须回返回一段JS片段，包含了服务器数据

doResponse({ename:"tom"});

客户端特意提供了一个名为doResponse(data)函数;

06\_jsonp.html http://localhost/06\_jsonp.html

06\_jsonp.php http://127.0.0.1/06\_jsonp.php

3:京东首页 登录 产品列表

