

浏览器缓存知识cookie

cookie(必须在服务器环境下):

自定义属性, 获取出来的都是字符串。

cookie体积在低版本浏览器下字字千金。

生命周期:

默认的生命周期为本次浏览器关闭之前;

不同的浏览器不共享cookie;

在获取cookie的时候一个cookie与另一个cookie是用***; ***分号加上空格

```
setCookie('sql', ' [{"小明":123456}, {"小凤":654321}] ', 1);
// setCookie('key', '小明', 1);
var onOff = true;
btn.onclick = function() {
    var uVal = user.value;
    var pVal = pass.value;
    var json = JSON.parse(getCookie('sql')); //把json变成数组
    for(var i=0; i<json.length; i++) {
        for(var attr in json[i]) {
            if(attr == uVal) { //说明有这个人
                //console.log(json[i][attr] , pVal)
                if(json[i][attr] == pVal) { //验证数据库的密码与填写的
                    密码匹配否

                    setCookie('key', uVal, 1); //如果匹配就设置key
                    window.location.reload(); //刷新页面
                    onOff = false; //区分有没有这个用户
                    return;
                } else {
                    alert('用户名密码错误!');
                    onOff = false;
                }
            }
        }
    }

    //如果既不是用户密码错误, 也没有正确匹配到, 说明没有这个人要注册。
    if(onOff) {
        alert('请注册!');
    }
}

//找key有没有被设置, 如果没有, 就要让他登录, 有就xxx欢迎来到
if(getCookie('key') != -1) {
    btn.value = getCookie('key') + '欢迎来到miaov! ~~';
    user.style.display = 'none';
    pass.style.display = 'none';
    btn.disabled = true;
} else {
```

```

    btn.value = '登录';
    user.style.display = 'block';
    pass.style.display = 'block';
    btn.disabled = false;
}
function getCookie(key) {
    var arr = document.cookie.split('; ');
    for(var i=0;i<arr.length;i++){
        var arr2 = arr[i].split('=');
        if(arr2[0] === key){
            return arr2[1];
        }
    }
    return -1;
}
function setCookie(key,value,time) {
    var odate = new Date();
    odate.setDate(odate.getDate()+time);
    document.cookie = key+'='+value+';expires='+ (time?odate: '');
}

```

```

-----
//setCookie('sql','[{"小明":123456},{"小凤":654321}]',1);
//    setCookie('key','小明',1);
var onOff = true;
btn.onclick = function(){
    var uVal = user.value;
    var pVal = pass.value;
    var json = JSON.parse(getCookie('sql')); //把json变成数组
    for(var i=0;i<json.length;i++){

        for(var attr in json[i]){

            if(attr == uVal){ //说明有这个人
                //console.log(json[i][attr] , pVal)
                if(json[i][attr] == pVal){ //验证数据库的密码与填写的
                    setCookie('key',uVal,1); //如果匹配就设置key
                    window.location.reload(); //刷新页面
                    onOff = false; //区分有没有这个用户
                    return;
                }else{
                    alert('用户名密码错误!');
                    onOff = false;
                }
            }
        }
    }
}

```

密码匹配否

//如果既不是用户密码错误，也没有正确匹配到，说明没有这个人要注册。

```

        if(onOff){
            alert('请注册!');
        }
    }
}

```

//注册:

```

var onOff2 = true;
btn2.onclick = function(){
    var uVal = user.value;
    var pVal = pass.value;
    var json = JSON.parse(getCookie('sql'));
    for(var i=0;i<json.length;i++){
        for(var attr in json[i]){
            if(attr == uVal){
                alert('你已经注册');
                onOff2 = false;
            }
        }
    }
}

```

```

if(onOff2){
    console.log(json)
    json.push({[uVal]:pVal});
}

```

setCookie('sql',JSON.stringify(json));//直接设置cookie值为对象，会默认调用toString

```

        this.style.display = 'none';
    }
}

```

```

btn3.onclick = function(){
    removeCookie('key',getCookie('key'));
    window.location.reload();
}

```

//找key有没有被设置，如果没有，就要让他登录，有就xxx欢迎来到

```

if(getCookie('key')!=-1){
    h1.innerHTML = getCookie('key')+'欢迎来到miaov! ~~';
    user.style.display = 'none';
    pass.style.display = 'none';
    btn2.style.display = 'none';
    btn3.style.display = 'inline-block';
    btn.disabled = true;
}else{
    btn.value = '登录';
    user.style.display = 'block';
}

```

```
pass.style.display = 'block';
btn3.style.display = 'none';
btn.disabled = false;
}

function getCookie(key) {
    var arr = document.cookie.split('; ');
    for(var i=0;i<arr.length;i++){
        var arr2 = arr[i].split('=');
        if(arr2[0] === key){
            return arr2[1];
        }
    }
    return -1;
}

function setCookie(key,value,time){
    var odate = new Date();
    odate.setDate(odate.getDate()+time);
    document.cookie = key+'='+value+';expires='+ (time?odate: '');
}

function removeCookie(key,value){
    setCookie(key,value,-2);
}
```