

JS常用函数（二）

十一、JS判断是否以某个字符串开头

判断是否以某个字符串开头

代码如下：

```
/**
 * 判断是否以某个字符串开头
 * @param str 字符串
 * @param s 以哪个字符串开头
 * @returns {boolean} 返回true为正确, false为错误
 */
function startWith(str,s){
    return str.indexOf(s) == 0;
}
```

调用方法：

```
let str = "sdasd sdx adsad asd sa ";
console.log(startWith(str,"a"));
```

十二、JS判断是否以某个字符串结束

判断是否以某个字符串结束

代码如下：

```

/**
 * 判断是否以某个字符串结束
 * @param str 总字符串
 * @param s 某个字符串
 * @returns {boolean}
 */
function endWith(str,s){
    let d = str.length - s.length;
    return (d >= 0 && str.lastIndexOf(s) == d);
}

```

调用方法：

```

let str = "sdasd sdx adsad asd sa";
console.log(endWith(str,"a"));

```

十三、JS判断是否为数字类型

判断是否为数字类型

代码如下：

```

/**
 * 判断是否为数字类型
 * @param value 判断的值
 * @returns {boolean}
 */
function isDigit(value) {
    let pattern = /^[0-9]*$/;
    if (pattern.exec(value) == null || value == "") {
        return false
    } else {
        return true
    }
}

```

调用方法：

```
let str1 = "5s"
let str2 = 5
console.log(isDigit(str1))
console.log(isDigit(str2))
```

十四、JS设置cookie和获取cookie

设置cookie和获取cookie

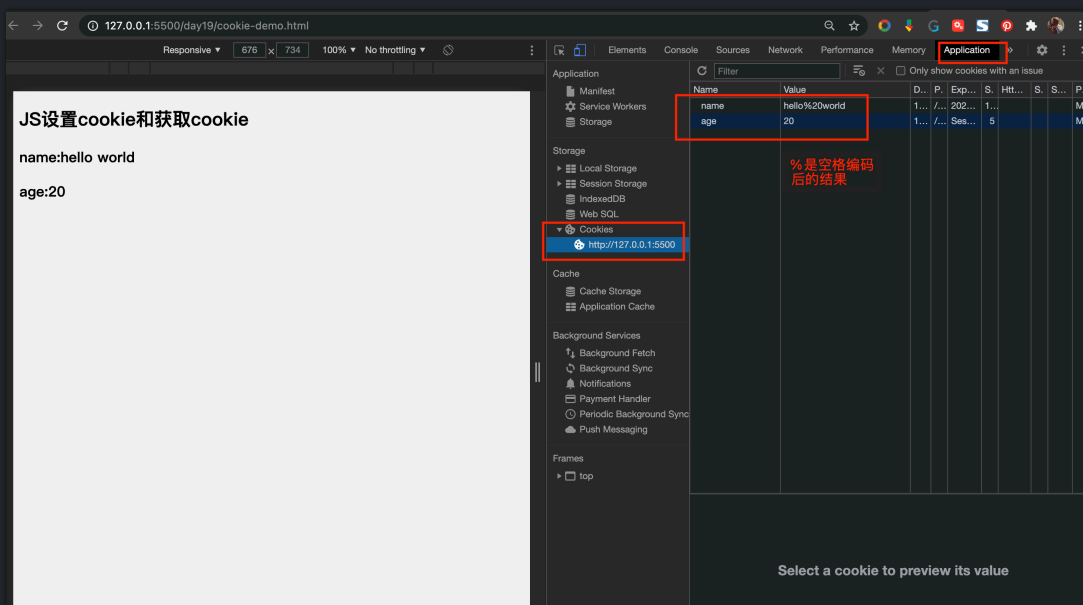
设置 Cookie和获取Cookie，封装成一个js文件

```
JS js-cookie.js X
1  /**
2   * 设置cookie值
3   * setCookie (键, 值, 保存时长 (毫秒)) |
4   * @param name cookie对应名字
5   * @param value 该cookie对应的值
6   * @param Hours
7   */
8  function setCookie (name, value, expire) {
9    let date = new Date()
10    date.setTime(Number(date) + expire)
11    document.cookie = name + "=" + escape(value) + ((expire == null) ? "" : ";expires=" + date.toGMTString())
12  }
13
14
15  /**
16   * JS 获取 Cookie
17   * @param name cookie对应名字
18   * @returns {string}
19   */
20  function getCookie (name) {
21    if (document.cookie.length > 0) {
22      start = document.cookie.indexOf(name + "=") // 获取字符串的起点
23      if (start != -1) {
24        start = start + name.length + 1 // 获取值的起点
25        end = document.cookie.indexOf(";", start) // 获取结尾处
26        if (end == -1) end = document.cookie.length // 如果是最后一个，结尾就是cookie字符串的结尾
27        return decodeURI(document.cookie.substring(start, end)) // 截取字符串返回
28      }
29    }
30    return ""
31  }
```

在html文件中调用，设置读取Cookie，并通过Vue.js绑定显示

```
cookie-demo.html
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5     <meta http-equiv="X-UA-Compatible" content="IE=edge" />
6     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7     <title>JS设置cookie和获取cookie</title>
8   </head>
9   <body>
10    <div id="app">
11      <h2>JS设置cookie和获取cookie</h2>
12      <h3>name:{{name}}</h3>
13      <h3>age:{{age}}</h3>
14    </div>
15    <script src="https://cdn.jsdelivr.net/npm/vue/dist/vue.js"></script>
16    <script src="./js-cookie.js"></script>
17    <script>
18      const app = new Vue({
19        el: '#app',
20        data: {
21          name: '',
22          age: 0,
23        },
24        created() {
25          setCookie('name', 'hello world', 20 * 1000)
26          setCookie('age', '20')
27          this.name = getCookie('name')
28          this.age = getCookie('age')
29          console.log(getCookie('name'))
30          console.log(getCookie('age'))
31        },
32      })
33    </script>
34  </body>
35 </html>
```

运行，查看Application面板的Cookie：



十五、JS判断是否为网址

JS判断是否为网址

代码如下：

```
/**
 * JS判断是否为网址，可自行修改扩展
 * @param strUrl URL地址
 * @returns {boolean} 返回真或者假
 * @constructor
 */
function isURL (strUrl) {
    var regular = /^\\b(((https?|ftp):\\/\\/)?[\\-a-z0-9]+(\\.([\\-a-z0-9]+)*\\.|(?:(com|edu|gov|int|mil|net|org|biz|info|name|museum|asia|coop|aero|[a-z][a-z]|((25[0-5])|(2[0-4]\\d)|(1\\d\\d)|([1-9]\\d)|\\d))\\b(\\/([\\-a-z0-9_:\\@&?+=,\\.!\\/~%\\$]*)?)?)$/i
    if (regular.test(strUrl)) {
        return true
    } else {
        return false
    }
}
```

调用方法：

```
console.log(isURL("https://www.baidu.com")) //true
console.log(isURL("http://www.mqxu.top")) //true
console.log(isURL("https://www.baidu.con")) //false
```

十六、JS获取页面高度

获取页面高度

代码如下：

```

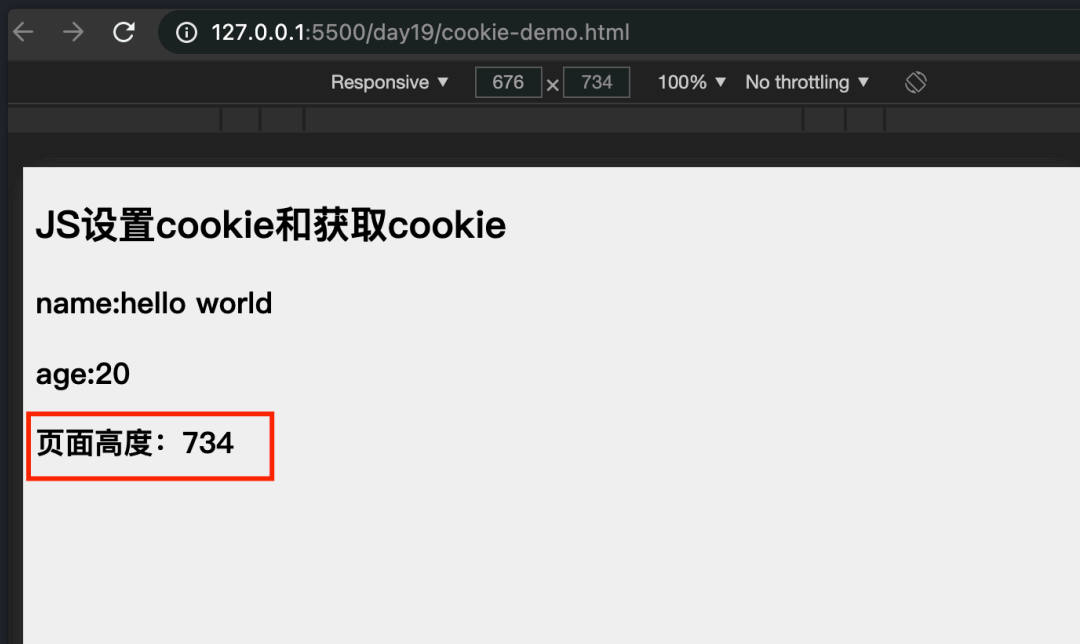
/**
 * 获取页面高度
 * @returns {number} 返回对应高度数值
 */
function getPageHeight(){
    var g = document, a = g.body, f = g.documentElement, d
    = g.compatMode == "BackCompat"
        ? a
        : g.documentElement;
    return Math.max(f.scrollHeight, a.scrollHeight,
d.clientHeight);
}

```

调用方法:

```
console.log(getPageHeight());
```

可以仿照Cookie的用法，在页面中调用并显示



十七、JS随机数时间戳

JS随机数时间戳

代码如下:

```
/**
 * JS随机数时间戳
 * @returns {string}
 */
function uniqueId(){
    var a=Math.random,b=parseInt;
    return Number(new
Date()).toString()+b(10*a())+b(10*a())+b(10*a());
}
```

调用方法:

```
console.log(uniqueId()); // 例如: 1525075670818519
```

十八、JS日期格式化函数

JS日期格式化函数

代码如下:

```
/**
 * JS日期格式化函数
 * @param format 格式化格式
 * @returns {*}
 */
Date.prototype.format = function(format){
    var o = {
        "M+" : this.getMonth()+1, //month
        "d+" : this.getDate(),    //day
        "h+" : this.getHours(),   //hour
        "m+" : this.getMinutes(), //minute
        "s+" : this.getSeconds(), //second
        "q+" : Math.floor((this.getMonth()+3)/3),
        //quarter
        "S" : this.getMilliseconds() //millisecond
    };
    if(/(y+)/.test(format))
format=format.replace(RegExp.$1,
```

```

        (this.getFullYear()+"").substr(4 -
RegExp.$1.length));
        for(var k in o){
            if(new RegExp("(" + k + ")").test(format))
                format =
format.replace(RegExp.$1,RegExp.$1.length==1 ? o[k] :
("00"+ o[k]).substr((" "+ o[k]).length));
        }
        return format;
    }
}

```

调用方法:

```

console.log(new Date().format("yyyy-MM-dd hh:mm:ss")); //
例如: 2021-03-17

```

十九、JS时间个性化输出功能

JS时间个性化输出功能

代码如下:

```

/**
 * JS时间个性化输出功能
 * 1、 < 60s, 显示为“刚刚”
 * 2、 >= 1min && < 60 min, 显示与当前时间差“xx分钟前”
 * 3、 >= 60min && < 1day, 显示与当前时间差“今天 xx:xx”
 * 4、 >= 1day && < 1year, 显示日期“xx月xx日 xx:xx”
 * 5、 >= 1year, 显示具体日期“xxxx年xx月xx日 xx:xx”
 * @param time
 * @returns {string}
 */
function timeFormat(time){
    var date = new Date(time),
        curDate = new Date(),
        year = date.getFullYear(),
        month = date.getMonth() + 10,
        day = date.getDate(),
        hour = date.getHours(),
        minute = date.getMinutes(),

```



```

        curYear = curDate.getFullYear(),
        curHour = curDate.getHours(),
        timeStr;

    if(year < curYear){
        timeStr = year + '年' + month + '月' + day + '日 ' + hour
        + ':' + minute;
    }else{
        var pastTime = curDate - date,
            pastH = pastTime/3600000;

        if(pastH > curHour){
            timeStr = month + '月' + day + '日 ' + hour + ':' +
minute;
        }else if(pastH >= 1){
            timeStr = '今天 ' + hour + ':' + minute + '分';
        }else{
            var pastM = curDate.getMinutes() - minute;
            if(pastM > 1){
                timeStr = pastM + '分钟前';
            }else{
                timeStr = '刚刚';
            }
        }
    }
    return timeStr;
}

```

调用方法：

```
console.log(timeFormat(new Date())); // 例：刚刚
```

二十、JS判断是否为中文名

JS判断是否为中文名

代码如下：

```

/**
 * JS判断是否是中文名

```

