

## JS第一周第五天

- 1.for of遍历
- 2.for循环
- 3.判断
- 4.获取元素的方法
- 5.获取和设置属性
- 6.选项卡
- 7.隔行变色

# JS第一周第五天

javascript

## 1.for of遍历

```
<body>
<ul id="list">
  <li>A</li>
  <li>B</li>
  <li>C</li>
</ul>
</body>

<script>
  //使用for of遍历数组
  let arr=["A","B","C"];
  for(let key in arr){
    //key:索引
    //arr[key]:数组的每一项
  }
  for (let item of arr){
    //item 数组的每一项
    console.log(item);
  }

  //"<li>A</li><li>B</li><li>C</li>"

  let str='';
  for (let item of arr){
    str+=`<li>${item}</li>`
  };
  list.innerHTML=str;
</script>
```

## 2.for循环

```
<script>
    for(var i=0;i<10;i++){
        console.log(i);
    }

    //for (语句1; 语句2; 语句3) {循环体}
    //1.语句1先执行，而且只执行一次
    //2.在执行语句2，是一个判断条件
    //3.如果语句2条件成立，执行循环体
    //4.循环体执行完之后，最后执行语句3
    //5.依次去循环2.3.4.步

    //语句3 是在循环体执行完之后执行的，所以可以直接将语句3放在循环体的最底下，但是语句2后面必须加上分号(;)

    //循环中的关键字：continue和break
    //continue 结束本次循环，进入下一循环
    //break 终止循环

    for (var i=0;i<7;i++){
        if(i==5)continue;//遇到continue，不继续往下执行，直接执行i++，进入下一次循环
        console.log(i);//0,1,2,3,4,6
    }

    for (var i=0;i<7;i++){
        if(i==5)break;//遇到break,直接终止
        console.log(i);//0,1,2,3,4
    }

    //找出从0到10的偶数
    for(var i=0;i<11;i++){
        if(i%2)continue;
        console.log(i);
    }

    //找出0到10中第一个被3整除的数
    for(var i=1;i<11;i++){
        if(i%3==0) {
            console.log(i);
            break;
        }
    }
}
```

```
</script>
```

### 3.判断

```
<script>
    var n=10;
    if(n>0&&n<=9){
        console.log("一位数");
    }else if(n>9&&n<=99){
        console.log("两位数");
    }else{
        console.log("多位数");
    }
}

//三元运算符
//条件? 条件成立执行的(只能一条语句):条件不成立执行的(只能一条语句);
var num=10;
if (num%2){
    console.log("奇数");
}else{
    console.log("偶数");
}

let m=null;
m=num%2?num:0;

//如果num是奇数, 让m="red" 如果num是偶数, 让m="green"

m=num%2?"red":"green";

switch (n){
    //switch 中的参数去跟case后面的值比较使用的是"==="
    case "a"://n==="a" 成立就会执行这里的代码
        console.log("a");
        break;
    case "b":
        console.log("b");
        break;
    default:
        console.log("哈哈");
        break;
}

if(true)console.log("你好");
//如果{}中只有一行代码, 可以省略{}
}
```

```
</script>
```

## 4. 获取元素的方法

```
<script>
    //通过ID名获取一个元素，如果有相同的ID名，获取的是第一个
    //document: 上下文
    //ID名的上下文只能是documents

    //通过类名获取一组元素 获取的是元素的集合，类数组（有length，有索引）
    //上下文.document.getElementsByClassName("类名")
    //上下文 自己选择，一般都是父级，爷爷也行，最好别超三辈
    //let pplist=document.getElementsByClassName("pp") 获取box下面所有叫pp
    的元素
    //可以通过索引找到对应的那一项 pplist[3].style.color
    //若只有一个类名
    let body=document.body;//获取body
    let box=body.getElementsByClassName("box")[0]
    //通过getElementsByClassName获取的肯定是一个集合，即使只有一个元素，如
    果想获取这个唯一的元素，需要通过索引[0]将其取出

    //通过标签名获取出来的也是一组元素集合，类数组（有length，有索引）
    //上下文.document.getElementsByTagName("标签名")
    box.getElementsByTagName("p");//获取box下的所有p标签

    //通过Name获取一组元素 只适用表单元素
    //上下文必须为document
    let nusers=document.getElementsByName("users");
</script>
```

## 5. 获取和设置属性

```
<script>
    //一个元素自带很多属性（例如id），叫做内置属性
    //我们也可以自己给他定义一些属性，叫做自定义属性
    //使用元素.自定义属性=值，不会显示在标签上（Elements下）
    let oDiv=document.body.getElementsByTagName("div")[0];
    oDiv.a="哈哈";
    console.dir(oDiv);

    //oDiv.setAttribute("属性名", 属性值) 给一个元素设置属性
```

```

//之前有就是修改，没有就是增加
//setAttribute 设置的属性会显示在标签上

//oDiv.getAttribute("属性名") 获取属性值
//可以获取内置属性，但是对于内置属性，只能获取通过setAttribute设置的（直接
在标签上写的自定义属性，相当于用setAttribute设置的，同样可以获取）

//oDiv.getAttribute("属性名") 删除属性
//可以移除内置属性。但是对于内置属性只能移除setAttribute设置的

//对于设置和获取的属性值都是字符串
</script>

```

## 6.选项卡

```

<body>
  <div id="banner">
    <div id="bannerInner">
      
      
      
      
      
    </div>
    <ul id="bannerTip">
      <li class="select">1</li>
      <li>2</li>
      <li>3</li>
      <li>4</li>
      <li>5</li>
    </ul>
    <a href="javascript:;" id="left"></a>
    <a href="javascript:;" id="right"></a>
  </div>
</body>

<script>
  // 获取元素
  let bannerInner=document.getElementById("bannerInner");
  let imgList=bannerInner.getElementsByTagName("img");
  let bannerTip=document.getElementById("bannerTip");
  let list=bannerTip.getElementsByTagName("li");
  let left=document.getElementById("left");
  let right=document.getElementById("right");

  let index=0;//记录当前显示的img的索引

```

```

//1.给所有的li绑定点击事件 循环绑定
for(let i=0;i<list.length;i++){
    list[i].onclick=function () {
        //记录显示的img索引
        index=i;

        //先让所有的li都没有类名，可以循环所有的li，让每一个li的className=""

        //先让所有的img没有这个select类名
        /* for (var j=0;j<list.length;j++){
            list[j].className="";
            imgList[j].className="";
        }
        //再让点击的那个li加上类名，跟li(索引)对应的img加上类名
        list[i].className="select";
        imgList[i].className="select";*/
        change(index)
    }
}

//给right绑定点击事件
right.onclick=function () {
    index++;
    //临界值判断，如果index大于4，再从0开始
    if(index>4)index=0;
    change(index)
}

left.onclick=function () {
    index--;
    //临界值判断，如果index大于4，再从0开始
    if(index<0)index=4;
    change(index)
}

function change(i){
    for (var j=0;j<list.length;j++){
        list[j].className="";
        imgList[j].className="";
    }
    list[i].className="select";
    imgList[i].className="select";
}
</script>

```

## 7.隔行变色

```
<body>
<ul id="list">
  <li>1</li>
  <li>2</li>
  <li>3</li>
  <li>4</li>
  <li>5</li>
  <li>6</li>
  <li>7</li>
  <li>8</li>
  <li>9</li>
</ul>
</body>

<script>
  let list=document.getElementById("list");
  let olis=list.getElementsByTagName("li");

  //给元素设置类名    元素.className=类名;
  for(var i=0;i<olis.length;i++){
    olis[i].className="li"+i%2;
  }

  //设置ID    元素名.id=id名

</script>
```