函数

1.1 函数概念

具有某种特定功能的代码块

1.2 函数的作用

可以解决代码复用问题

将整个项目模块化

1.3 函数的定义方式

1. 方式1 函数声明方式 function 关键字 (命名函数)

```
function  函数名([arg1[,arg2...]]){
    代码块
    return 返回值;
}
函数名(实参列表);//调用
```

2. 方式2 函数表达式或者函数字面量 (匿名函数)

    1. 语法

```
var 变量=function(参数){
    return  返回值
}

变量名(实参); //调用
这个地方的函数实际上是一个匿名函数
```

    注意:

    (1) 在函数被定义时，函数内部的代码不会执行

    (2) 在函数被调用时，函数内部的代码会被执行

1.4 定义函数要素

1. 功能

   一般说的就是函数能做什么，通常情况下函数名就代表着函数的功能(见名识意)

2. 参数

   函数的小括号当中，写的是参数，函数定义的参数，我们被称作形参（形式参数)

   函数调用的时候,小括号里面也写的是参数,这些参数我们称作实参(实际 参数);

   函数调用的实参本质上是在给函数定义的形参 进行赋值(把实参的值赋值给形参，通常叫传参)

   形参列表 必须 与实参列表保持一致

3. 返回值

   每一个函数都会有返回值，有的函数返回值比较明确，写了return

有的函数没有写return，并不是代表这个函数没有返回值，而是省略了 return，实际上返回值是 undefined

每个函数只能有一个返回值

1.5函数分类

1. 无参 无返回值

```
function fun1(){
    console.log("我要学习");
}
fun1();
```

2. 无参 有返回值

3. 有参 无返回值

```
function fun2(num1,num2){
    var result = num1+num2;
    console.log(result);
}

fun2(100,200);//调用
```

4. 有参 有返回值

```
function fun2(num1,num2){
    var result = num1+num2;
    //返回值
    return result;
}

var res = fun2(50,50);//调用
console.log(res);
```

1.5.函数练习 使用函数表达式 函数声明式 都完成一遍

1、求圆的周长 2*pair*（r半径）

```
function circleL(r){
    var l = 2*3.14*r;
    return l;
}
var res1 = circleL(3);
console.log("圆的周长为"+res1);
```

2、求圆的面积　pairr

3、把一个字符串翻转

```
/**
 * 把一个字符串翻转
 * @param str String
 * @return new_str
 */
function strrev(str){
    var new_str = "";
    for(var i=str.length-1;i>=0;i--){
        new_str +=str[i];
    }
    return new_str;
}
var res = strrev("abcd");
console.log(res);
```

4、把一个数组翻转

1.6函数参数

1. 如果形参列表、实参列表不一致 则出现错误
2. arguments对象
    1. JavaScript 函数有一个名为 arguments 对象的内置对象
    2. arguments 对象包含函数调用时使用的参数数组。

       当不清楚要有多少个参数时才使用，可以使用函数进行查找

       

1.7匿名函数

1. 匿名函数就是没有名字的函数
2. 匿名函数通常与自执行函数结合使用，因为匿名函数没有函数名，没办法调用，通过自执行调用
3. 匿名函数的基本形式
    1. （function(){}）()

```
(function(){
       console.log("我是匿名函数");
})();
```

    2. (function(){}())

```
(function(){
   console.log("我是你匿名函数 我加一个括号 使自己调用自己");

}());
```

1.8回调函数

1. 作用

   js代码会至上而下一条线执行下去，但是有时候我们需要等到一个操作结束之后再进行下一个操作，这时候就需要用到回调函数

2. 解释

   1. 因为函数实际上是一种对象，它可以存储在变量中，通过参数传递给另一个函数，在函数内部创建，从函数中返回结果值”，因为函数是内置对象，我们可以将它作为参数传递给另一个函数，到函数中执行，甚至执行后将它返回
   2. 回调函数就是一个参数，将这个函数作为参数传到另一个函数里面，当那个函数执行完之后，再执行传进去的这个函数。这个过程就叫做回调

3. 用法

```
function show(arg,fun2){

     //show函数 本身要执行的代码

    console.log(arg);



    //调用另一个函数

    fun2(20);

  }



  function show2(age){

    console.log("年龄为"+age)

  }



  show("参数",show2);
```