函数定义形式

2020年03月02日, 星期一　　08:52

1.函数是什么

函数是可以通过外部代码调用的一个“子程序”。

一个函数由称为函数体的一系列语句组成。值可以传递给一个函数，函数将返回一个值。

函数定义方式

通过**函数声明**的形式来定义

通过**函数表达式**的形式来定义

通过 Function 构造函数实例化的形式来定义

注意：函数声明与函数表达式的区别是：function关键字前是否有其他内容

2.通过 Function 构造函数创建函数

- 可以传入任意数量的实参
- 最后一个实参为函数体
- 函数体中 javascript 语句之间分号隔开
- Function 构造函数创建一个匿名函数

new Function(**[arg1[, arg2[, ...argN]],]**, functionBody)

```
var max = new Function("a", "b", "return a > b ? a : b;");
```

3.函数定义三要素

- 函数名：如 alert、 parseInt 、……
- 函数的参数：传递给函数名的值，代表将被函数处理的数据
- 函数的返回值：函数执行的返回结果

4.匿名函数（没有函数名）

- 单独的匿名函数是无法运行的
- 可以把匿名函数赋值给变量或立即执行

5.具名函数（有函数名）

- 当遇到错误时，堆栈跟踪会显示函数名，容易寻找错误

```
function f1() {
    (function() {
        console.error("error");
    })();
}
f1();
```

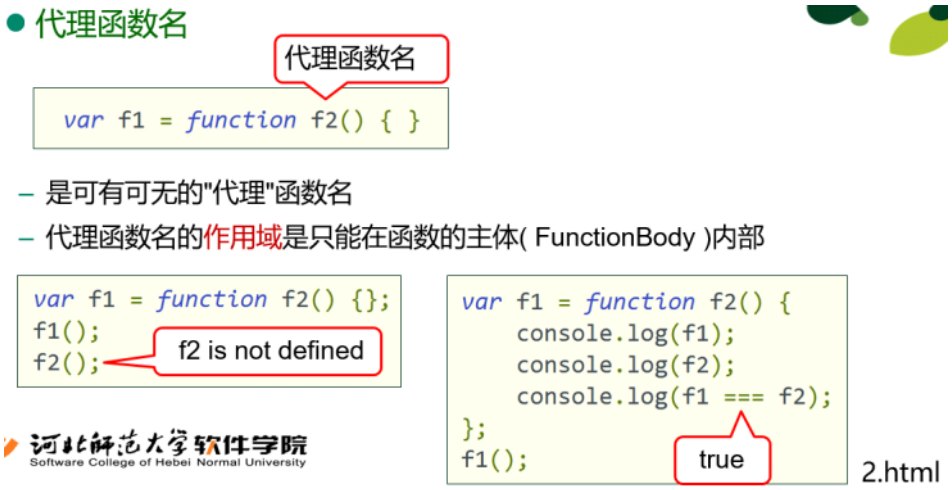
```
function f1() {
  (function() {
    console.error("error");
  })();
}
f1();
```

```
error
    (anonymous) @ VM336:3
    f1          @ VM336:4
    (anonymous) @ VM336:6
undefined
```

```
function f1() {
  (function f2() {
    console.error("error");
  })();
}
f1();
```

```
error
    f2          @ VM348:3
    f1          @ VM348:4
    (anonymous) @ VM348:6
undefined
```

6.代理函数名




```
var f1 = function f2() {};
f1();

undefined
var f1 = function f2() {};

f2();
Uncaught ReferenceError: f2 is not defined
    at <anonymous>:3:7
```

```
var f1 = function f2() {
  console.log(f1);
  console.log(f2);
  console.log(f1 === f2);
};
f1();
```

```
ƒ f2() {
  console.log(f1);
  console.log(f2);
  console.log(f1 === f2);
}
```

```
> var f1 = function f2() {
    console.log(f1);
    console.log(f2);
    console.log(f1 === f2);
  };
  f1();
ƒ f2() {
    console.log(f1);
    console.log(f2);
    console.log(f1 === f2);
  }
ƒ f2() {
    console.log(f1);
    console.log(f2);
    console.log(f1 === f2);
  }
true
undefined
>
```

f1和f2指向的是同一地址

7.name属性

- 返回函数实例的名称

```
> var f1 = function f2() {
    console.log(f1.name);
    console.log(f2.name);
  };
  f1();
f2
f2
```

```
var f1 = function() {};
console.log(f1.name);

var obj = { method: function() {} };
console.log(obj.method.name);

var f1 = new Function();
console.log(f1.name);
```

使用 new Function()语法的函数其名称为“anonymous”

```
> var f1 = function() {};
  console.log(f1.name);

  var obj = { method: function() {} };
  console.log(obj.method.name);

  var f1 = new Function();
  console.log(f1.name);
f1
method
anonymous
```

8.length属性

- length 属性指明函数定义的形参个数

```
function func1() {}
console.log(func1.length);
// output: 0

function func2(a, b) {}
console.log(func2.length);
// output: 2
```