Technology Tarena 达内科技

Function 对象

创建函数

- JavaScript中创建函数的三种方式
  - 使用function关键字声明方式创建命名函数
  - 使用直接量赋值方式创建命名函数
  - 使用Function构造方法创建函数

知识讲解

Technology Tarena 达内科技

创建函数

- 使用声明方式创建函数:

```
function  函数名( 形参1, 形参2, ... ){
    函数体
    return 返回值
};
```

- 问题: 会被声明提前

知识讲解

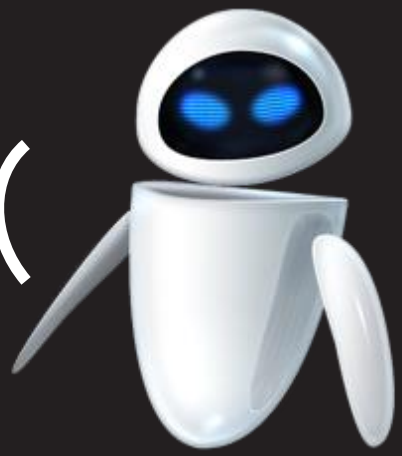
Technology Tarena 达内科技

创建函数

- 回顾: 声明提前(hoist)——在js程序开始执行前，引擎会查找所有var声明的变量和function声明的函数，集中到当前作用域的顶部集中创建。赋值留在原地。

知识讲解

```
{
    console.log(i);
    var i=10;
    console.log(i);
}
```



创建函数

- 回顾: 声明提前(hoist)——在js程序开始执行前，引擎会查找所有var声明的变量和function声明的函数，集中到当前作用域的顶部集中创建。赋值留在原地。

知识讲解

```
{                                    hoist
        ; //undefined
    console.log(i); //undefined
    var i=10;
    console.log(i); //10

}
```

Technology Tarena 达内科技

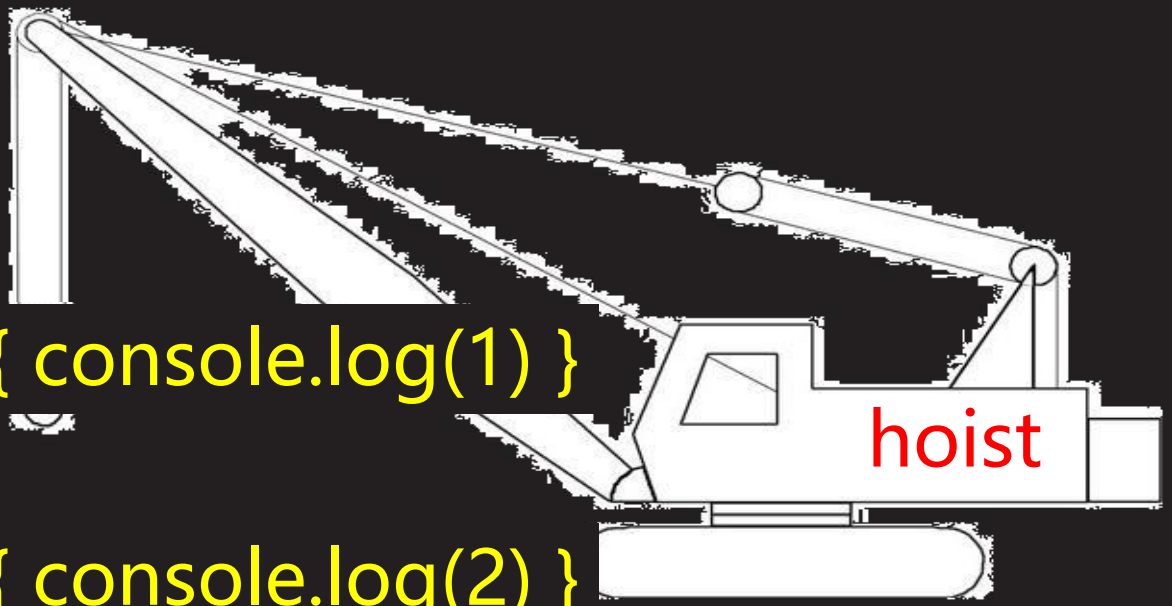
创建函数

- 重现声明提前问题：
  - console.log(i);
  - var i=10;
  - console.log(i);
  - 问：以上程序是否可正常执行，如果可以正常执行，输出结果是什么？如果不能正常执行，则报什么错?
  - 答案: 可以正常执行，输出： undefined 10

课堂练习

创建函数

- 回顾: 声明提前(hoist)——在js程序开始执行前，引擎会查找所有var声明的变量和function声明的函数，集中到当前作用域的顶部集中创建。赋值留在原地。




```
function fun(){ console.log(1) }
fun(); //输出?
function fun(){ console.log(2) }
fun(); //输出?
```

知识讲解

Technology Tarena 达内科技

创建函数

- 重现声明提前问题：
  - function fun(){ console.log(1) }
  - fun();
  - function fun(){ console.log(2) }
  - fun();
  - 问：以上程序输出结果是?
  - 答案: 2 2

课堂练习

创建函数

- 声明提前是JS一个广受诟病的缺陷，因为破坏了程序正常的执行顺序，极易产生歧义
- 如何避免函数声明提前：改声明方式为赋值方式创建函数
- 使用赋值方式创建函数：

```
var    function 函数名 (形参1,形参2,...){
         函数体;
         return 返回值;
       }
```

- 因为是赋值方式，所以不会被声明提前，按程序书写顺序执行

知识讲解

Technology Tarena 达内科技

创建函数

- 重现声明提前问题：
  - var fun=function(){ console.log(1) }
  - fun();
  - var fun=function(){ console.log(2) }
  - fun();
  - 问：以上程序输出结果是?
  - 答案: 1 2

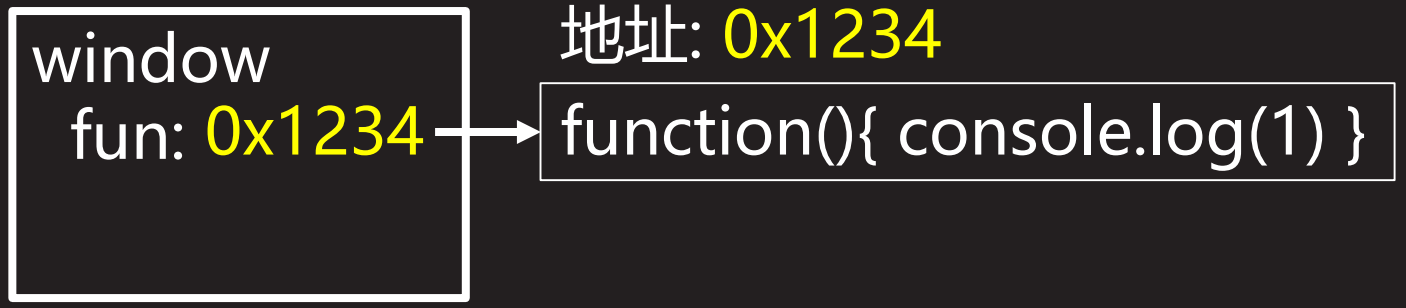
课堂练习

创建函数

Technology Tarena 达内科技

知识讲解

- JS中函数的本质:
  - 1. 函数其实是一个保存一段代码的对象
  - 2. 函数名其实只是一个普通的变量。函数名变量通过对象地址引用了一个函数对象，所以使用函数名变量等效于使用函数对象
- 比如:
  - 无论: function fun(){ console.log(1) }
  - 还是: var fun=function(){ console.log(1) }
  - 实际上都是:



Technology Tarena 达内科技

创建函数

- 重现声明提前问题：
  - var fun=function(){ console.log(1) }
  - fun();
  - var fun=function(){ console.log(2) }
  - fun();
  - var fun=100;
  - console.log(fun);
  - fun();
  - 问：以上程序输出结果是?
  - 答案: 1 2 100 报错: fun不是一个函数

课堂练习

创建函数

- 使用 Function 对象直接创建函数

  var 函数名=
  new Function(“形参1”，“形参2”,...，“函数体”)

- 问题: 代码可读性差，效率低，所以很少用

知识讲解

Technology Tarena 达内科技