## Day09\_javaScript基础

### 0.1学习目标

1、能够掌握js基本语法

2、能够掌握js的基本数据类型

3、能够掌握js常用的对象

### 0.2 学习指南

* Js基本语法（重要）
* Js基本数据类型（重要）
* Js常用对象（重点）

### 第一堂课

本节知识点

1、今日内容

2、JavaScript\_简介

3、JavaScript\_发展史

4、JavaScript\_语法\_与html结合方式

本节目标

js的概述和发展史

#### 今日内容

##### 1.1.1知识概述

今天主要是js基础

##### 1.1.2视频详情



##### 1.1.3总结与补充

无

##### 1.1.4课堂提问与练习

##### 1.1.5习题答案

##### 1.1.6视频缺陷

无

##### 1.1.7视频扩展

无

#### JavaScript\_简介

##### 1.2.1知识概述

 javascript是什么,一种客户端脚本语法,作用与html交互。

##### 1.2.2视频详情



##### 1.2.3总结与补充

##### 1.2.4课堂提问与练习

什么是javaScript？

##### 1.2.5习题答案

Javascript:客户端脚本语言,作用与html交互,让页面具有动态效果,增强用户体验

##### 1.2.6视频缺陷

无

##### 1.2.7视频扩展

无

#### JavaScript\_发展史

##### 1.3.1知识概述

通过javaScript的发展历史,要清楚javascript具体有哪些部分组成

##### 1.3.2视频详情



##### 1.3.3总结与补充

javascript经过不断的发展,基本组成部分有bom对象,dom对象,以及基本语法组成。

##### 1.3.4课堂提问与练习

##### 1.3.5习题答案

##### 1.3.6视频缺陷

无

##### 1.3.7视频扩展

无

#### JavaScript\_语法\_与html结合方式

##### 1.4.1知识概述

要想让html具有动态效果,需要引入javascript文件,引入方式和css基本一样

##### 1.4.2视频详情



##### 1.1.3总结与补充

##### 1.4.4课堂提问与练习

##### 1.4.5习题答案

##### 1.4.6视频缺陷

无

##### 1.4.7视频扩展

无

### 第二堂课

本节知识点：

1、JavaScript\_语法\_注释&数据类型

2、JavaScript\_语法\_变量

3、JavaScript\_语法\_变量\_typeof

4、JavaScript\_语法\_一元运算符

本节目标 ：

js的基本语法和定义变量

#### JavaScript\_语法\_注释&数据类型

##### 2.1.1知识概述

js的基本语法,注释基本和java的语法一样的,数据类型稍微有些差别

##### 2.1.2视频详情



##### 2.1.3总结与补充

##### 2.1.4课堂提问与练习

* 五种原始类型有哪些？

##### 2.1.5习题答案

Number

String

Null

Boolean

undefined

##### 2.1.6视频缺陷

无

##### 2.2.7视频扩展

无

#### JavaScript\_语法\_变量

##### 2.2.1知识概述

js是一门弱类型脚本语法,所有的定义变量都用var接收

##### 2.2.2视频详情



##### 2.2.3总结与补充

js是弱类型客户端编程语言,区分和强类型编程语言的区别.

##### 2.2.4课堂提问与练习

##### 2.2.5习题答案

##### 2.2.6视频缺陷

无

##### 2.2.7视频扩展

无

#### JavaScript\_语法\_变量\_typeof

##### 2.3.1知识概述

在java里面可以使用instanceof 来判断类型,在js里面可以使用typeof判断变量的类型

##### 2.3.2视频详情



##### 2.3.3总结与补充

##### 2.3.4课堂提问与练习

无

##### 2.3.5习题答案

无

##### 2.3.6视频缺陷

无

##### 2.3.7视频扩展

无

#### JavaScript\_语法\_一元运算符

##### 2.4.1知识概述

在js里进行运算,要区分清楚字符串和数字之间的加减运算.

##### 2.4.2视频详情



##### 2.4.3总结与补

##### 2.4.4课堂提问与练习

无

##### 2.4.5习题答案

无

##### 2.4.6视频缺陷

无

##### 2.4.7视频扩展

无

### 第三堂课

本节知识点

1、JavaScript\_语法\_算数&比较运算符

2、JavaScript\_语法\_逻辑运算符

3、JavaScript\_语法\_三元运算符

4、JavaScript\_语法\_特殊语法

本节目标

掌握js运算符

#### JavaScript\_语法\_算数&比较运算符

##### 3.1.1知识概述

js里面进行运算,加减乘除,带有小数点的 要和java区分清楚,比较运算符==,!=,>=,<=等

##### 3.1.2视频详情



##### 3.1.3总结与补充

##### 3.1.4课堂提问与练习

##### 3.1.5习题答案

##### 3.1.6视频缺陷

无

##### 3.1.7视频扩展

无

#### JavaScript\_语法\_逻辑运算符

##### 3.2.1知识概述

js里面常见的逻辑运算符,有&& ,|| ,!

##### 3.2.2视频详情



##### 3.2.3总结与补充

其实js里面的逻辑算符的用法和java是一样的

##### 3.2.4课堂提问与练习

##### 3.2.5习题答案

##### 3.2.6视频缺陷

无

##### 3.2.7视频扩展

无

#### JavaScript\_语法\_三元运算符

##### 3.3.1知识概述

js的三元运算符和java里面的是一样的,用途很广.

##### 3.3.2视频详情



##### 3.3.3总结与补充

##### 3.3.4课堂提问与练习

##### 3.3.5习题答案

##### 3.3.6视频缺陷

无

##### 3.3.7视频扩展

无

#### JavaScript\_语法\_特殊语法

##### 3.4.1知识概述

js里面的定义变量,决定是全局变量,还是局部变量,取决于变量的位置.

##### 3.4.2视频详情



##### 3.4.3总结与补充

##### 3.4.4课堂提问与练习

* 练习js的运算符

##### 3.4.5习题答案

参见:代码目录下,一元.逻辑.比较运算符等

##### 3.3.6视频缺陷

变量是否是全局变量或者局部变量,取决于变量的位置,而不是有没有定义var这个关键字

比如:

function demo(){ b=4;}

alert(b);

如果没有调用demo这个方法,会报错: 错误信息是b未定义,

##### 3.3.7视频扩展

### 第四堂课

本节知识点：

1、JavaScript\_语法\_流程控制语句

2、JavaScript\_语法\_练习\_99乘法表

3、JavaScript\_对象\_Function

本节目标 ：

掌握创建函数的几种方式

#### JavaScript\_语法\_流程控制语句

##### 4.1.1知识概述

js的流程控制语句有if, if ,else , switch case, while等

##### 4.1.2视频详情



##### 4.1.3总结与补充

##### 4.1.4课堂提问与练习

##### 4.1.5习题答案

##### 4.1.6视频缺陷

##### 4.1.7视频扩展

#### JavaScript\_语法\_练习\_99乘法表

##### 4.2.1知识概述

通过流程控制语句,连个for循环,打印99乘法表

##### 4.2.2视频详情



##### 4.2.3总结与补充

##### 4.2.4课堂提问与练习

* 练习99乘法表

##### 4.2.5习题答案

<script>

document.write("<table align='center'>");

//1.完成基本的for循环嵌套，展示乘法表

for (var i = 1; i <= 9 ; i++) {

document.write("<tr>");

for (var j = 1; j <=i ; j++) {

document.write("<td>");

//输出 1 \* 1 = 1

document.write(i + " \* " + j + " = " + ( i\*j) +"&nbsp;&nbsp;&nbsp;");

document.write("</td>");

}

/\*//输出换行

document.write("<br>");\*/

document.write("</tr>");

}

//2.完成表格嵌套

document.write("</table>");

</script>

##### 4.2.6视频缺陷

##### 4.2.7视频扩展

#### JavaScript\_对象\_Function

##### 4.3.1知识概述

js里面创建函数有三种方式,大家要学会第一种和第二种的创建方式.

##### 4.3.2视频详情



##### 4.3.3总结与补充

##### 4.3.4课堂提问与练习

* 练习创建函数的几种方式

##### 4.2.5习题答案

//1.创建方式1

var fun1 = new Function("a","b","c","alert(a);");

//调用方法

// fun1(3,4);

// alert(fun1.length);

//2. 创建方式2

function fun2(a , b){

alert(a + b);

}

//fun2(3,4);

//alert(fun2.length);

var fun3 = function(a,b){

alert(a+b);

}

//alert(fun3.length);

// fun3(3,4);

/\*fun2 = function(a , b){

alert(a - b);

}\*/

function fun2(a , b){

//alert(a - b);

alert(a);

alert(b);

}

//fun2(3,4);

//方法调用

//fun2(1,2);

//fun2(1);

//fun2();

//fun2(1,2,3);

/\*

\* 求两个数的和

\*/

/\*function add(a , b){

return a + b;

}\*/

/\*\*

\* 求任意个数的和

\*/

function add (){

var sum = 0;

for (var i = 0; i < arguments.length; i++) {

sum += arguments[i];

}

return sum;

}

var sum = add(1,2,3,4);

alert(sum);

//add(1,3);4.2.6视频缺陷

无

##### 4.2.6视频缺陷

##### 4.2.7视频扩展

无

### 第五堂课

本节知识点：

1、JavaScript\_对象\_Array

2、JavaScript\_对象\_Date

3、JavaScript\_对象\_Math

本节目标 ：

掌握Aarry,Date,Math对象的使用

#### JavaScript\_对象\_Array

##### 5.1.1知识概述

在js里面要想使用数组,首先创建数组对象,创建方式大体有四种,js里面数组的长度不是固定的.

##### 5.1.2视频详情



##### 5.1.3总结与补充

##### 5.1.4课堂提问与练习

* 练习数组对象常用方法和属性

##### 5.1.5习题答案

var arr = [1,"abc",true];

document.write(arr +"<br>");

document.write(arr[0] +"<br>");

document.write(arr[1] +"<br>");

document.write(arr[2] +"<br>");

//document.write(arr[10] +"<br>");

arr[10] = "hehe";

document.write(arr[10] +"<br>");

document.write(arr[9] +"<br>");

//alert(arr.length);//11

document.write(arr.join("--")+"<br>");

arr.push(11);

document.write(arr.join("--")+"<br>");

##### 5.1.6视频缺陷

##### 5.1.7视频扩展

#### JavaScript\_对象\_Date

##### 5.2.1知识概述

Date对象,是日期对象,在js里面同样获取系统时间.

##### 5.2.2视频详情



##### 5.2.3总结与补充

##### 5.2.4课堂提问与练习

* 练习Date对象的方法

##### 5.2.5习题答案

var date = new Date();

document.write(date + "<br>");

document.write(date.toLocaleString() + "<br>");

document.write(date.getTime() + "<br>");

##### 5.2.6视频缺陷

无

##### 5.2.7视频扩展

无

#### JavaScript\_对象\_Math

##### 5.3.1知识概述

 Math对象可以进行获取随机数,四舍五入,向上取整和向下取整的方法.

##### 5.3.2视频详情



##### 5.3.3总结与补充

Math：数学对象

1. 创建：

\* 特点：Math对象不用创建，直接使用。 Math.方法名();

2. 方法：

random():返回 0 ~ 1 之间的随机数。 含0不含1

ceil(x)：对数进行上舍入。

floor(x)：对数进行下舍入。

round(x)：把数四舍五入为最接近的整数。

3. 属性：

PI

##### 5.3.4课堂提问与练习

* 练习Math对象的方法

##### 5.3.5习题答案

document.write(Math.PI +"<br>");

document.write(Math.random() +"<br>");

document.write(Math.round(3.14) +"<br>");

document.write(Math.ceil(3.14) +"<br>");

document.write(Math.floor(3.14) +"<br>");

/\*\*

\* 取 1~100之间的随机整数

\* 1. Math.random()产生随机数：范围 [0,1)小数

\* 2. 乘以 100 --> [0,99.9999] 小数

\* 3. 舍弃小数部分 ：floor --> [0,99] 整数

\* 4. +1 -->[0,99] 整数 [1,100]

\*

\*

\*/

var number = Math.floor((Math.random() \* 100)) + 1;

document.write(number);

##### 5.3.6视频缺陷

无

##### 5.3.7视频扩展

无

### 第六堂课

本节知识点：

1. JavaScript\_对象\_RegExp1
2. JavaScript\_对象\_RegExp2
3. JavaScript\_对象\_RegExp3
4. JavaScript\_对象\_Global

本节目标 ：

掌握正则表达式对象的使用,正则表达式不需要大家熟练写出来

#### JavaScript\_对象\_RegExp1

##### 6.1.1知识概述

大家要想使用RegExp对象,首先清楚正则表达式怎么书写

##### 6.1.2视频详情



##### 6.1.3总结与补充

##### 6.1.4课堂提问与练习

##### 6.1.5习题答案

##### 6.1.6视频缺陷

无

##### 6.1.7视频扩展

无

#### JavaScript\_对象\_RegExp2

##### 6.2.1知识概述

正则表达式对象,就是为了校验用户输入的值是否符合规范

##### 6.2.2视频详情



##### 6.1.3总结与补充

##### 6.2.4课堂提问与练习

##### 6.2.5习题答案

##### 6.2.6视频缺陷

无

##### 6.2.7视频扩展

无

#### JavaScript\_对象\_RegExp3

##### 6.3.1知识概述

正则表达式对象,基本上应用于用户名,密码,手机,邮箱等的校验操作

##### 6.3.2视频详情



##### 6.3.3总结与补充

##### 6.3.4课堂提问与练习

* 练习正则表达式对象的方法

##### 6.3.5习题答案

var reg = new RegExp("^\\w{6,12}$");

//2.

var reg2= /^\w{6,12}$/;

/\*alert(reg);

alert(reg2);\*/

var username = "zhangsan";

//验证

var flag = reg.test(username);

alert(flag);

##### 6.3.6视频缺陷

##### 6.3.7视频扩展

#### JavaScript\_对象\_Global

##### 6.4.1知识概述

js里面的全局对象,就是可以不用创建对象,可以直接拿来使用,所以在调用全局对象 的方法时,可以不用写对象

##### 6.4.2视频详情



##### 6.4.3总结与补充

无

##### 6.4.4课堂提问与练习

##### 6.4.5习题答

##### 6.4.6视频缺陷

无

##### 6.4.7视频扩展

无