

前端知识体系复盘-第二阶段

一、JavaScript基础概念

(1)数据类型

了解计算机基础：

计算机运行代码的时候。

内存、CPU、硬盘

内存：负责报错你们运行的数据，一旦关闭电脑内存清空。

CPU：负责高速的运算，将结果保存内存

硬盘：数据持久化

```
let a = 10
let j = 20
let sum = a + j
console.log(a)
JSON.stringify(a)
```

计算机最终只能识别二进制数据。写程序用的高级编程语言。

任务程序写完了，都需要经过编译、解释、转化为二进制、交给计算机运行

V8引擎：运行你的代码，将这段代码解释为机器语言。

```
let a = 10
a = 10 //内存分类一个空间 名字a 这个空间存放1010
j = 20 //
+ ==>转化二进制
```

发现有+让cpu执行运算

内存产生一个sum空间，cpu运算结果

log是一个函数，浏览器封装的函数。

数据类型：

实际上定义了内存的空间大小。

每个变量都会保存内存中，但是用多大的内容来保存，开发者说了算。

比如Java代码

```
int a = 20 //int代表内存分配4字节
long j = 30 //内存分配8字节
```

总结：

基本数据类型：number、boolean、string、null、undefined、Symbol、bigint

引用类型：对象类型

完善：

1. JS代码运行在浏览器中，我要研究一下 let a = 20
2. Number到底最大能存多大的数字

(2)关于数据转化

强类型语言：数据转化一般显示的转换。

弱类型语言：默认转换和强制转换

默认（隐式）转换：

```
let i = 10;  
let j = "20"  
let sum = i - j // -10
```

在执行这段代码的时候，如果遇到 `-*/%==` 默认进行类型转化

将字符串类型默认转化为数字类型执行运算

显示转化：

字符串转数字：parseInt、parseFloat、Number

数字转字符串：拼接、JSON.stringify String()

```
orderId:20230203  
orderId.indexOf(2023) //报错 indexOf is not function
```

解决方案：

```
const newOrderId = orderId+""  
newOrderId.indexOf("2023")  
const res = JSON.stringify(orderId)  
  
String(orderId)
```

字符串转布尔：

```
Boolean("")  
Boolean("1")
```

布尔类型转数字：

```
Number(true) //1
```

扩展：

1. 数据类型默认根据值来决定的。值一旦确认了内存空间大小就定了
2. 官网还根估计基本数据类型提供包装类型

number---Number

boolean---Boolean

包装类型就算基本数据类型升级，用面向对象类和对象来表示你们内存的数据

```
Number(10)
let num = 10
```

```
function(){
  for(){
    for(){
      return;
    }
  }
}
```

(3)函数的定义

JS中函数定义两种：

函数(申明)定义：

```
function show(){

}
```

函数表达式：

```
const show = function(){

}
```

箭头函数

```
const play = ()=>{

}
```

参数：形参和实参

返回值：返回结果

学会查看方法签名

```
let array = [1,2,3,4]
const boo = array.some((item)=>{

})
```

(4)常见api操作

数组的API:

添加：push、

删除：splice、pop

修改：splice

查询: forEach、map、filter、some、every

字符串api:

新增: 拼接、下标

删除: 替换

修改: 下标 "abc"[0] = n

替换: replace\replaceAll

查询:

截取: substr、substring split

```
abc    abd mk    cbdhbcd
```

```
split(/s+/)
```

(5)内置对象

在JS中，浏览器默认内置了很多对象，可以直接供开发使用。无需定义

```
Date  
Array  
String  
RegExp  
Boolean  
Math  
localStorage  
Object
```

Nodejs:相当于将chrome的V8引擎核心代码提取出来，单独设计了一套JS运行环境，

JS运行在Nodejs里面没有浏览器各种安全限制。

JS调用Nodejs提供的api，操作本地操作系统或者硬盘

要求:

将内置对象的api整理一下

二、WebAPI

学习DOM操作

DOM编程并不是我们JS提出的概念。

DOM是一种独立编程规范，至少JS中用了这种规范。

剩下API:

```
document.getElementById ("app")
```

和

```
const res = document.querySelector(".app")
```

`[...res].map()`

动态的节点和静态节点（伪数组）

NodeList、HTMLCollection

思考：测试动态和静态