今天学习的内容

1:复习上周知识重点

2:作业项目---(重点&难点)

3:今天的目标

3.1:WebWorker--代码就3行,重点在理论理解

进程:操作系统分配内存的单位--工厂

线程:进程内部，用于执行代码--生产线

线程**并发**:操作系统中所有的线程宏观上看"同时执行"

微观上看是"依次交替执行"

chrome中线程的模型：请求资源-6个线程;运行代码/渲染内容--1线程(UI主线程)

<button>按钮1</button>

<script src="1.js"></script>

<button>按钮2</button>

现象:JS执行过程中，按钮1可见，按钮2不可见

原因:浏览器执行JS和渲染网页是一个线程

解决一:将所有js程序执行放在body最后一个元素

解决二:创建新的线程，由它来执行耗时js任务,

UI主线程执行后续的HTML渲染

<button>按钮1</button>

<script>

var w = new Worker("1.js");

</script>

<button>按钮2</button>

3.2:WebWorker--缺陷

浏览器不允许Worker线程操作DOM&BOM对象

原因: 浏览器只允许UI主线程操作DOM/BOM,

若多个Worker线程同时操作DOM结构，页面

混乱...

**Woker线程可以给UI主线发送数据**

UI主线程:

var w6 = new Worker("1.js");

w6.onmesssage = function(e){e.data}

Worker线程:

postMessage(stringMsg);

**UI线程可以给Worker主线发送数据**

UI主线程:

var w6 = new Worker("1.js");

w6. postMessage(stringMsg);

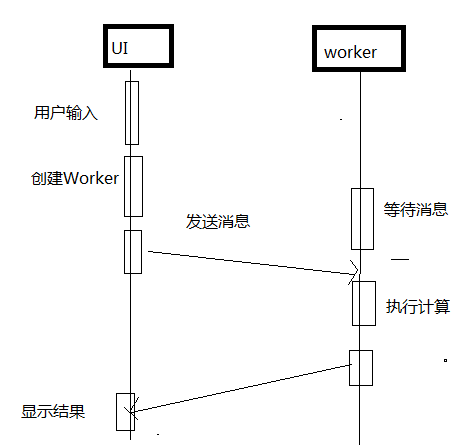
Worker线程:

onmesssage = function(e){e.data}

练习:在HTML中有一个INPUT元素，按钮"开始计算"

,点击此按钮，创建一个Worker线程来计算用户输入

数字累加和,在下方div显示出计算结果



作业1:产品管理(添加产品)