

函数定义和调用过程

js是编译性语言

- js是具有函数优先的轻量级、解释型或及时编译型的编程语言
- 编译型语言
 - 写的程序执行之前需要一个专门的编译过程，把所有的程序编译成为机器语言的文件
- 解释型语言
 - 在运行程序的时候才编译，每一执行一段代码就要翻译一段代码
- 解释性语言和编译性语言的不同是编译的时机不同

js代码的执行过程

Js引擎是一段一段的运行代码的；js代码执行时会为当前代码创建相应的运行环境

js运行环境

- 全局环境：代码运行起来后，会进入全局环境
- 函数环境：当函数被调用执行时会进入当前函数中的执行代码
- eval环境：不建议使用

执行上下文

- 执行上下文
 - 可以理解为当前代码的运行环境，作用是用来保存当前代码运行时所需要的数据，在全局环境函数环境中会创建执行上下文
- 执行上下文栈
 - 按照函数的调用顺序来管理执行上下文，栈底永远是全局上下文，栈顶是当前正在执行的函数
 - 特点：先进后出
- 全局上下文什么时候出栈或者销毁？关闭页面退出程序
- 函数执行时才会创建执行上下文
- 执行上下文的生命周期
 - 创建阶段
 - 生成变量对象
 - 建立作用域链
 - 确定this指向
 - 执行阶段
 - 变量赋值
 - 函数引用
 - 执行其他代码
 - 执行完毕后，出栈等待被回收
- 每一个执行上下文都有一个与之关联的变量对象和作用域链
- return语句的作用：返回值；终止函数执行（即销毁当前执行上下文，弹出执行上下文栈）

变量对象

- 变量对象
 - 变量对象是与执行上下文相关的数据作用域，变量对象是与上下文关联的特殊对象，用于存储被定义在上下文中的变量和函数声明。
- 变量对象创建过程
 - 建立arguments对象,检查当前上下文中的函数声明，检查当前上下文中的变量声明
- 代码运行机制
 - 代码编译
 - 将代码翻译成可执行代码
 - 代码执行阶段
 - 执行可执行代码
- js编译和执行过程
 - Js编译和执行是一个不断交替的过程
 - 全局编译阶段（预解析）
 - 全局顺序执行阶段（变量赋值，函数调用等操作）
 - 当遇到函数调用时，在执行函数内代码时进行函数范围内的编译
 - 当存在函数嵌套时，以此类推会进行多次函数预解析