**es6 day04**

1. **模块化（面试重点）**

es5不支持模块化的

es6出来，才有模块化，需要放到服务器的环境才行

如果定义一个模块？

a.js

const a=12;

export 东西;(导出或者暴露出来)

b.js

import “./a.js”

import {a} from “./a.js”

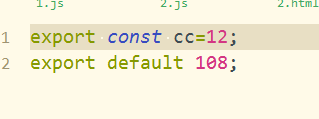
import的特点：

1. import 后面可以是相对路径也可以是绝对路径

import “http://4490.replace.favo.njb97.com/”

1. import模块只能导入一次
2. import “./a.js”相当于引入了a.js
3. import 有提升的效果，他可以提升到代码的顶部

如果是export导出，import就必须加{ }



1. html--1.js(模块)--2.js(模块)

import 可以动态的引入（能够获取最新的值）

import(“路径”)

1. 按需加载（）
2. 可以在if语句里面
3. 路径可以是动态的
4. import() 返回的是一个promise对象
5. html 引入了100个模块（没有用按需加载）--100个文件都要加载---1.html文件会加载非常慢，可能会出现白屏
6. html 用了按需加载，模块用的时候才加载。
7. 类和继承（考试重点）

es6里面如何定义一个类

注意：

1. es6里面class没有提升的功能
2. es6里面关于this比之前轻松多了

es6类里面如何定义静态的方法？就是这个类自身的方法

语法 static aaa(){

}

继承：

extends

语法 class Student extends Person{

}

super进行解释（非常重要，面试的重点内容）

super可以作为对象，也可以作为方法。

1. super作为函数时，代表父类的构造函数。

虽然super代表父类的构造函数，但此时返回的时B的实例，即super内部的this指的是B的实例

因此super()相当于 A.prototype.constructor.call(this)

2.super作为对象时，在普通方法中，指向父类的原型对象，在静态方法中，指向父类

// super 作为对象，在静态方法中使用，相当于父类，而不是父类的原型

// super作为对象 ，在静态方法使用，super==Parent

// super作为对象，在实例方法使用 super==Parent.prototype

https://blog.csdn.net/u013448372/article/details/111997556