# 基础

java的执行过程

标识符 变量常量

二进制 16进制 之间的转换

类型 8 种 基本类型

整数 小数 布尔

类型 自动转换 强制装换 同种类型之间可以

包装类的使用 不同种类型

包的定义和使用

运算符 算术，逻辑，赋值

控制语句

数组的使用

方法 函数 重载（条件）主要跟参数有关

参数 可变参数

返回值 参数以及返回值类型

参数类型如果是基本类型 实参和形参之间的关系

如果是引用类型，他们之间的关系又是如何的

git的使用

提交以及获取

add(加到缓存)->commit(本地仓库)->push(远程仓库)

fetch(获取到本地仓库)-ckeckout或者merge 合并，到工作区

pull 直接从远程拉取到工作区。

生成SSH公钥，增加远程 连接

remote add origin git地址

git 版本库 github 远程库 gitee 国内的远程库

这周五计划 vue项目答辩

大家安装 腾讯会议 软件。

你们演示和演讲，大学中 毕业设计答辩

你们演示，我提问几个

目前为止大家项目的完成率

统计

# 面向对象

面向过程主要是方法的定义和调用

各种方法要知道在哪个文件中

而且你导入一次，使用的方法都是同一个

a.txt

全部的变量

show(){

局部的变量

}

include a.txt

调用这个方法了

show()

每个调用的地方使用的是同一个全局

调用的地方也有这个同名的变量，导致冲突等一些问题

面向过程编程的约束性比较大。

a.java

全局变量

show(){

局部变量

}

在每个地方调用的时候

构造对象 对象与对象之间是相互独立的，互不影响

A a=new A();

类中的全局 局部变量互不影响，new

a 对象了

对了一个对象的概念

A 类 就是文件

A a2=new A();

a对象和a2对象是相互独立了。（没有过多的约束性）

对象.的方式调用里面变量，方法等（符合人类思维方式）

面向过程

面向对象

以上两个是开发模式 大的方向

面向组件

面向切面

面向服务

以上三个是演变过来的开发模式 局部的定义

1.文件

2.类

3.对象

在文件中以类的形式书写代码

class 类 类型 和基本类型没有本质区别

他们都是类型

int

long

float

文件中定义的类 自定义类

int 只能存放数字

float只能存放小数

自定义类，里面可以存放很多东西，包括变量以及函数

关键字

class 定义 A类

A a;

int a=100 栈就可以 存的数据比较单一

class Int类

Int a=new Int() 存的数据就复杂了，而且很多，只能存在堆中

栈和堆

类只是一种定义，要使用就要实例化对象

int a int 基本类型 a 变量 栈

Int a Int 类 （自定义类型） a 对象（变量 堆）

本质上是没有区别

类 和 对象 官方说明

以上是我个人理解 没有官方

int a 好理解

类 对象 之间的关系

类模板，类别 动物类 人类

对象 个体 动物类中一条狗 旺财 （对象）

人类中的一个人，

张三（对象）

李四（对象）

张三和李四之间互不影响

类就是模板

写 –类

用-对象

类中主要是定义变量和方法

变量可以多个

方法也可以多个

变量-属性

方法-函数

成员 属性

成员 函数

成员就是对象的意思

static 不叫成员

通过.操作符，调用对象中的成员的。属性和方法

在java所有的文件都是类的形式定义

钢琴 学音乐

电子琴

小提琴

吉他

古筝

编程语言

最好还是先学c语言 面向过程 代表

java语言 面向对象 代表

js脚本语言 脚本语言代表

sql 语言 数据库 代表

构造函数

一般情况默认是没有值的。

要使用之前需要初始化

但是在类中是可以的

基本类型默认值 int 0 false 0.0

引用类型 null

自定定义一个构造函数

没有自己定义，怎么能使用呢？？

特点， 名字和类型一致

没有返回类型

自动调用

new 开辟空间，实例化对象。其实就是调用一个构造函数，来实例化对象的

本质上就是调用函数，就是因为这个函数，才会初始化对象中的属性的

通过调试查看

java类的执行顺序

面试的时候经常被问到

先调用初始化为默认值

再调用构造函数

调用构造函数代码之前再调用初始化

new Dog() 进一步分析，做了两件事情

开辟一个空间出现一个对象

调用构造函数

返回对象的引用

Dog d1=**new** Dog();

1.实例化对象，初始化默认值 int 0 引用 null

2.调用构造函数 之前 初始化 属性值

3.调用构造函数中的代码 代码中有属性赋值，再替换属性值

实例化对象，一般在构造函数中初始化值，还有不会初始化值在调用的地方赋值、

类和对象的执行过程

this代表的就是当前对象

虚拟机就是模拟真机一种软件 cpu 内存 硬盘 寄存器 指令系统等等。虚拟机也有这一套完善的体系结构

方法体中定义的变量是局部变量，生效时间从调用方法开始到结束，变量空间也会随之销毁

本地就是java调用c++程序，c++调用系统

native

进入函数调试F5 跳出函数调试 F7

f5 f6 f7 f8

参数类型如果是引用类型，则形参概念，会影响实参

他们就是同一个东西

包装类型就是对基本类型的一个包装变成引用类型。提供了更多的方法

基本类型只能存数据，包装类型可以通过方法操作

包装类单作基本类型赋值

实际上是new 一个对象的

基本类型->包装类型->引用类型

i1++

++ 是对基本类型的操作

引用是不行的

i1 引用

包装类型就是引用类型

i1++

首先把i1自动转换为基本类型

才可以++

Integer i2=i1;  
i1自动转为引用类型

jvm给我们做了这个自动转换 1.5包括 之后

装箱 拆箱

把基本类型自动包装成引用类型

反之 拆箱

i1++

Integer i2=**new** Integer(i1.intValue()+1);

i1++

产生一个新的对象了。

引用类型，改变形参，实参不受影响

包装类型的引用类型

包装类型可以单作基本类型使用。

对包装类型数字 ，有一个缓存 数字大小就是byte类型

常量池中

缓存数据的

自动装箱和拆箱，但是耗资源

int a=100

Integer a=new Integer(100);

int a= new Integer(100);

Integer a=100;

this就是一个变量名字。比较特殊 代表的意思就是是当前对象的一个引用

用法2