学一个知识点，就立马搞懂？

知识点之间是有关联的

java基本的工作

创建类

类概念还不知道

public static class main String[] args

java程序 源码.java

通过javac 编译器 编译一下变成.class字节码

java 虚拟加载class文件并且执行 找到入口函数

main函数

先讲类

先了解，后面回头复习 总结一下

接口 目的

什么是接口

怎么定义的

里面属性特点

函数是什么特点

怎么使用的

实现类 接口找中的所有抽象函数

1.8特性 等等

为何需要接口

通过传参

没有接口也是可以的，但是要定义很多重载方法

标准 参数中只要定义接口，只要定义一个方法即可

接口使用起来要方便 从设计角度考虑

功能一样，设计不一样，效率也就不一样

开发 功能

设计 全局

程序员 实现功能的 10

系统工程师 设计 20

架构师 项目的稳定性，可行性，安全性 40

技术专家 50 专门研究一个领域 攻克技术难题 交管监控系统 车牌图片识别成数字 算法 一开始没有 ，研究并且实现

疫情期间 疫苗 研究 生物技术领域

管理区分

5个月

达到很高层面去开发项目

需要项目开发经验

需要一个过程

疫苗 3-5年临床试验

学到后面越难了

难 并不是技术不理解

正常

接口 配合工厂模式 反射机制 一定项目经验 领悟到接口作用

jdk中有一些接口 方法定义都没有一个

空接口

上层使用都是基于这个接口 标准化

实现不是jdk做的，也遵守这个接口

java.sql包

数据库处理的

jdk 中 只有定义，没有任何实现 的

都是个数据库各大厂商去实现

mysql sqlserver oracle db2 等等 数据库

java对接 接口 标准

各大厂商按照这个标注去实现

java中可以使用各大厂商的数据库类库  
以后切换数据库，java代码不需要改任何，只要修改驱动 改数据库连接地址 ，账号和密码 即可

现在不讲 ，不行

后面 集合框架

基于 接口

collection

list

set

map

接口目前

继承

特点很明显

每个类都要定义重复的属性

重复的属性定义在基类中

extends继承这个基类，子类就拥有积累的属性和方法

有一些要求的

构造函数不能继承

封装 访问修饰

private 修饰 继承中也是不能被继承的

单继承

方法重载

方法重写

封装 使用 作为参数 作用

开发 都要考虑 封装

面向对象设计的基础 OOP

继承 项目小的话基本也不用

没有几个类

但是 在使用java中的其他技术的说，要用到继承

window窗体程序

靠继承来实现很多功能，而且不需要你自己写过多的代码

定义一只笔

设计粗细

设置颜色

设置形状 线

设置起始点。然后开始画

到结束点

java中只要继承jframe就可以实现了

对象开发 oop思想

对象中的方法，属性 设置一下这个窗体就出来了

t.setTitle("窗体程序");

t.setBounds(100, 100, 450, 300);

怎么理解OOP

java基础

基本 OOP 高级

网络编程 servlet技术

不会写网络程序

通讯行业

聊天程序

servlet 是重点

通过tomcat 封装了网络编程的功能

servlet业务实现

socket serversocket 监听 客户端请求

发送消息 接受消息

http协议 nodes讲过

主次

基础 OOP 核心类库 高级部分 图形化（作为学习还是需要学的）

java核心编程卷I 卷II

后面微服务 分布式 以前是不讲的

讲图形化

现在又讲了 总时间是不变的

30家 1家用已经不错

BS界面开发

就要考虑使用封装 OOP 思想

# 多态

针对方法来讲，属性没有多态

类中只有属性和方法

一个类中的一个方法

按理一个方法应该就一个实现。

调用一个方法的时候，有多种表现形态

animal.jiaosheng() 调用

狗叫

猫叫

其它动物叫

静态多态其实就是重载

show()

只要是参数不一样

调用的是同一个方法的时候，有不同的功能实现

重载 严格意义上说 不同方法

静态 编译的时候

动态 运行的时候

异常 编译型异常 运行时异常

网页图片

jpg(静) 静态网页？

gif(动) 动态网页？

并不是

静 编译阶段

动 运行阶段

静 数据是固定

动 数据是生成

重载 在一个类中也能实现

两个以上也能实现的

重载 静态多态

重写 动态多态

继承

方法重写

向上转型

才能实现多态

重写

继承

方法名一致，参数一致，返回类型一致

都一样才能重写

复写

建议 重写 最好都是一致的。

传递参数的时候可以用父类引用子类

在方法中把这个父类赋值给子类

子类 a=（子类）父类

编译没有问题，运行的时候看他真正的类型

传递什么类型就是什么类型

编译的时候是允许的，但是在运行的时候，发生类型不一致，则报错

后面无法访问的

父类引用子类。

子类强制转换

接口引用实现类

父类引用子类

调用的是属性，调用父类属性 左边

调用的是方法，调用的子类方法 右边

属性隐藏 方法多态

重写以及引用

原则：父类不能转换为子类，但是在某种条件下，是可以的，因为传过来的就是这个类型，只是通过强制转换一下而已

基本类型

低-》高 自动

高-》低 强制 大不了精度丢失，但是可以运行的

引用类型中

上 自动

下 强制（不允许的）

子类不能引用父类对象

多态的实现

并不是一个独立的新技术

而是有继承，重写，引用 之间的关系。

调用多个对象提现出来的形态 多态

OOP面向对象编程的基础部分

# 枚举

定义和使用

为何要使用枚举？

java开发中 ，只要一个类 基本类型 控制语句

接口 抽象类 继承 封装 等

枚举类 其实也是一个类 特殊

枚举类不能继承别的类

因为jdk 做了一些处理，让你定义的枚举类底层继承一个了Enum类，不能继承其它类了

ctrl+shift+t

快速打开一个指定的类型的源码 type

重写 tostring equals

private 构造函数

书写枚举 ，再学一个其它语法了

类

接口

抽象类

为什么这个语法可以呢

public class A{

}

1+1=2?

toString 它是object中的方法

equals 也是object中的方法

System.***out***.println 函数 参数

调用参数中对象的toString 方法

所有的类都是继承Object

toString

equals

自定义的类型，父类中的equals 还是判断的是两个对象的内存地址是否相同