**学JAVA的第一天之class的研究**

（1）.class文件和Java文件

1.class文件中存储的是JAVA类文件转换而成的二进制的字节码

2..Java文件中含有一个类或接口，则只有一个.class文件

3.含有多个内部类或接口，内部类的class文件名为：外部类名+$+内部类名

4.类中含有多个并行类，均以并行类命名

public class A{

class B{ }

interface C{

public void D ( );

}

} //有三个.class文件。A.class，A$B.class，A$C.class

（2）静态变量和非静态变量

1.非静态变量是一个对象的属性，需创建对象来调用，这样才会给这个对象分配空间，才能去调用属性。

2.静态变量不属于某个对象，是属于类，可以直接由类名调用。

class A{

public static String str；

private static String str1；

public String str2；

}

public class B{

public void test(){

A.str; //因为str是静态的，且为public修饰，所以可以被类A直接调用

//str1虽然是静态的，但是被private修饰，所以不能被其他类调用

A a = new A(); //str2是非静态的，必须要获得他的实例对象才能调用

a.str2;

}

}

（3）类的加载顺序

1.父类子类的加载顺序：先顶级类，再逐层往下

public class Person {

static{

System.out.println("Person prepare");

}

}

class Teacher extends Person{

static {

System.out.println("Teacher prepare");

}

}

class Bootstrap{

public static void main(String [] args) //更改启动类

{

new Teacher();

}

}

//运行结果为：

Person prepare

Teacher prepare

2.引用类的加载：只有当引用类被初始化了，才会加载这个引用类

public static Course course=new Course(); //初始化了的引用类

public static Course course; //未被初始化的引用类

（4）核心类Clas

Class对象是用来创建类的常规对象的，当我们编译一个Java类就会同时产生一个与该类相对应的Class对象(保存在.class文件中,即.java文件编译后的.class文件中)。该Class对象中保存了创建对象所需的所有信息。

A a = new A();

if(a.getClass()==A.class) {

System.out.println("equal");

} else {

System.out.println("unequal");

}

//结果为：equal

//对象a是A的一个实例，A是某一个类，在if语句中使用a.getClass()返回的结果正是类A的类型类，

在Java中表示一个特定类型的类型类可以用“类型.class”的方式获得，因为a.getClass()获得是A的类型类，也就是A.class