---------------------------------------------------

计划课程安排：

FUNDAMENTAL(7天)         该阶段已结束

OOP(11天)           该阶段已结束

JAVASE01(10天) 该阶段已结束

JAVASE02(12天) 该阶段已结束

WEBBASIC(7天) ←当前为该阶段第6天

JQUERY(2天)

DATABASE(6天)

JDBC(3天)

SERVLETJSP(12天)

SPRINGMYBATIS01(8天)

AJAX(2天)

SPRINGMYBATIS02(12天)

LINUX(2天)

NGINX(2天)

REDIS(2天)

T-STORE(5天)

预计合计：102天

预计剩余：57天

注：上述信息随时可能发生更改，仅供参考。

---------------------------------------------------

**2019年5月6日**

**Java就业班-第55天**

**8:30实训室2-早自习**

今日晨讲主题——金属材料（？？？）

今天晨讲的第二组组长，也是班上唯一的女性学员。

然后她的晨讲内容与程序无关，据其说是因为java学的太菜，所以晨讲就不讲java相关内容了。

晨讲的内容是其大学专业的知识，主要以材料化学为主，包括一些实验报告、曲线图等。

理所当然地全班人包括项目经理都听得一脸懵逼。

大概5分钟就讲完了，也就是随便应付了一下吧，顺便透露了其是南京工业大学的毕业生。

……

然后项目经理表示今天的不作数，明天重新讲。

**9:00实训室2-正课**

开始上课。

又是开始回顾昨天课程内容……

**10:12实训室2-正课**

不行了不行了，我听不下去了，第6天了居然还在讲标签，还是自行看视频学java去，不然时间都给浪费了。

**10:15实训室2-正课**

关于Annotation标注。(先了解，不深入)

（以下部分文字内容摘抄自百度）

Annontation是Java5开始引入的新特征。中文名称一般叫注解。它提供了一种安全的类似注释的机制，用来将任何的信息或元数据（metadata）与程序元素（类、方法、成员变量等）进行关联。

annotation一般作为一种辅助途径，应用在软件框架或工具中，在这些工具类中根据不同的 annontation注解信息采取不同的处理过程或改变相应程序元素（类、方法及成员变量等）的行为。

例如：Junit、Struts、Spring等流行工具框架中均广泛使用了annontion。使代码的灵活性大提高。

从java5版本开始，自带了三种标准annontation类型：

1.@Override -在方法或类上标注时，表示此方法重写了父类方法。

java.lang.Override 是一个marker annotation类型，它被用作标注方法。它说明了被标注的方法重载了父类的方法，起到了断言的作用。如果我们使用了这种annotation在一个没有覆盖父类方法的方法时，java编译器将以一个编译错误来警示。

测试代码：

class Person{

public String info() {

return "这是一个Person类";

}

}

class Student extends Person{

@Override

public String inFo() { //此处刻意将方法名写错

return "这是一个Student类";

}

}

public class OverrideDemo01 {

public static void main(String[] args) {

Person p1 = new Student();

System.out.println(p1.info());

}

}

这个annotaton常常在我们试图覆盖父类方法而确又写错了方法名时加一个保障性的校验过程。

2.@Deprecated –在方法或类上标注时，编译时会提示此方法或类不建议使用。

Deprecated也是一种marker annotation。当一个类型或者类型成员使用@Deprecated修饰的话，编译器将不鼓励使用这个被标注的程序元素。所以使用这种修饰具有一定的 “延续性”：如果我们在代码中通过继承或者覆盖的方式使用了这个过时的类型或者成员，虽然继承或者覆盖后的类型或者成员并不是被声明为@Deprecated，但编译器仍然要报警。

测试代码：

*@Deprecated* // 声明不建议使用的操作

class ~~Human~~{

public String info() {

return "这是一个Human类";

}

}

public class DeprecatedDemo01 {

public static void main(String[] args) {

~~Human~~ p1 = new ~~Human~~();

System.***out***.println(p1.~~info~~());

}

}

注意：@Deprecated这个annotation类型和javadoc中的 @deprecated这个tag是有区别的：前者是java编译器识别的，而后者是被javadoc工具所识别用来生成文档（包含程序成员为什么已经过时、它应当如何被禁止或者替代的描述）。

3.@SuppressWarning -此注解能告诉Java编译器关闭对类、方法及成员变量的警告。

有时编译时会提出一些警告，对于这些警告有的隐藏着Bug，有的是无法避免的，对于某些不想看到的警告信息，可以通过这个注解来屏蔽。

测试代码：

class Demo<T>{

private T var ;

public T getVar(){

return this.var ;

}

public void setVar(T var){

this.var = var ;

}

};

public class SuppressWarningsAnnotationDemo01{

@SuppressWarnings("unchecked")

public static void main(String args[]){

Demo d = new Demo() ;

d.setVar("Kamui") ;

System.out.println("内容：" + d.getVar()) ;

}

};

SuppressWarning不是一个marker annotation。它有一个类型为String[]的成员，这个成员的值为被禁止的警告名。对于javac编译器来讲，被-Xlint选项有效的警告名也同样对@SuppressWarings有效，同时编译器忽略掉无法识别的警告名。

**11:21实训室2-正课**

简单了解一下数据库编程。

关于JDBC。

JDBC（Java Database Connectivity,即Java数据库连接）是指提供了一种与平台无关的用于执行SQL语句的标准Java API。

可以方便的实现多种关系型数据库的统一操作。它由一组用Java语言编写的类和接口组成。

※数据库的操作过程：

·打开数据库的服务

·连接数据库，连接的时候一般都要输入用户名和密码。

·操作数据库：创建表、查询表、更新记录。

·关闭

※数据库的操作：Statement、PreparedStatement

※数据库的查询:ResultSet

※调用存储过程：CallableStatement

JDBC的主要操作类及接口：（java.sql.\*）

DriverManager：是一个最常用的类，使用此类可以取得一个数据库的连接。

Connection:每一个Connection的实例化对象都表示一个数据库连接。

…………

**11:43实训室2-正课**

在知道了JDBC的作用之后再简单了解一下MySQL数据库。

MySQL是一个小型关系型数据库管理系统。主要特点为：体积小、速度快、总体拥有成本低、源码开放。

**11:50实训室2-午休时间**

下课了，出去晃晃。

**14:05实训室2-正课**

上课了，讲师下达了一项任务，班内分成几个小组，每组5个人合力制作一个网页，要求无限制，时限暂定为周三上午12点之前，完成后由项目经理选出1-2个优秀作品提交至机构总部讲师邮箱。现在开始制作。

今天下午似乎是不再讲课了，明天上课则正常讲新内容，所以实际上制作时间算上晚自习也就是1天的时间。

**15:46实训室2-正课**

然而在我观察了一轮之后发现大部分人都是在搞自己的事情，包括我也是……反正网页我是懒得做了，倒是更愿意预习一下后面要学的内容。

然而我还没从午休的疲惫中缓过来，所以我上各大招聘平台逛了逛，当做是做一下调查，了解了一下现在各公司招人的要求依然还是那么高。除了高要求的以外，其他的要么是一些培训机构伪装的，要么就是一些人力资源外包的公司。行业不景气的现象依旧体现的淋漓尽致啊。

……

等会再接着上午没看完的数据库内容继续往下预习吧。

**20:30实训室2-正课**

…………

不知不觉睡着了。

大概下午5点半左右的时候吧，刚学到MySQL数据库的简单增删改查的时候，感觉眼睛有点疲惫于是就趴桌上闭眼休息。醒来的时候发现睡过去2个半小时，与其说是睡过去，更像是昏睡过去了……因为在我的记忆里这2个半小时我还一直在安装配置数据库……直到被班主任叫醒，才发现我是在做梦。

也是被问到是怎么回事，为什么有那么多次晚自习都是在睡觉。我把我的情况简单的说明了一下，随后班主任表示如果晚自习的这两个小确实时影响到了我的日常精神状态，那么在保证学习进度跟得上的情况下，则可以不用强制参加晚自习。

对于不用强制上晚自习这一点我还是蛮认同的，只不过我的精神状态并不是因为上晚自习影响的，而是多年的老毛病导致的，简单点说就是因为失眠导致的嗜睡，看了一下日期发现这个毛病至今为止也已经有8年的历史了。因为平时清醒的时候觉得影响不是很严重，也没有因为这个症状造成过很严重的后果，所以对策方法最多也就是从医生那里开了一些安眠药，在需要的时候助睡一下。后来嫌每次药吃完需要再开太麻烦，往后助睡的方法就改成了饮酒……

算了，这个话题就不细说了。

一天又是这么浪费过去了，想要保持每天都处于一种高效学习的状态看来没有那么容易。翻了翻自己最开始写的日记，那时候的状态，在和现在做一下对比就发现，我真是变得越来越捞了。从经济状况上来说明明留给自己的时间不多了，我的危机意识却反倒是越来越薄弱了，现在满脑子想得都是什么时候我能真正睡个舒服的、安稳的觉，能够一觉睡到自然醒而不会睡得超过10个小时，能够像一个正常人一样一躺就着而不是必须得把脑子强制关机……

……

根据讲师下午所说的，明天开始应该就是新的课程内容了。简直不敢相信这个标签语言居然讲了足足6天！

……讲都讲了，只能罢了。明天再继续加把劲吧。

今日感想总结：Good night.

今日评分：-1/10分

今天就到这里，明天继续投稿。

感谢支持！

================================================================

**Acer程序员群：790482850**

**up主的java学习日记word文件及全部代码（随日记同步更新）：**

Adress：github.com/wt62635