笔记

2016年

[笔记 1](#_Toc465667038)

[1 笔记 5](#_Toc465667039)

[1.1 Java 5](#_Toc465667040)

[1.1.1 JDK 5](#_Toc465667041)

[1.1.2 JRE 5](#_Toc465667042)

[1.1.3 JVM 5](#_Toc465667043)

[1.1.4 Instanceof 5](#_Toc465667044)

[1.1.5 keytool 6](#_Toc465667045)

[1.1.6 序列化和反序列化 6](#_Toc465667046)

[1.2 将项目部署在linux服务器 7](#_Toc465667047)

[1.2.1 Linux命令大全 7](#_Toc465667048)

[1.2.2 部署项目 7](#_Toc465667049)

[1.3 Scheduler:定时器 12](#_Toc465667050)

[1.3.1 应用场景 12](#_Toc465667051)

[1.3.2 使用 12](#_Toc465667052)

[1.4 数据库 13](#_Toc465667053)

[1.4.1 删除数据 13](#_Toc465667054)

[1.4.2 数据库索引 13](#_Toc465667055)

[1.4.3 数据库引擎 14](#_Toc465667056)

[1.5 SSM 15](#_Toc465667057)

[1.5.1 搭建环境 15](#_Toc465667058)

[1.6 反射 15](#_Toc465667059)

[1.6.1 相关讲解 15](#_Toc465667060)

[1.6.2 运用 15](#_Toc465667061)

[1.7 Dom4j解析xml文件 15](#_Toc465667062)

[1.8 高并发 16](#_Toc465667063)

[1.8.1 耦合性 16](#_Toc465667064)

[1.8.2 内聚性 16](#_Toc465667065)

[1.9 Mysql 16](#_Toc465667066)

[1.9.1 内连接 16](#_Toc465667067)

[1.9.2 SELECT LAST\_INSERT\_ID() 17](#_Toc465667068)

[1.9.3 DATE\_SUB() 17](#_Toc465667069)

[1.9.4 Explain 18](#_Toc465667070)

[1.9.5 SELECT IFNULL(expr1,expr2) 18](#_Toc465667071)

[1.10 Redis 18](#_Toc465667072)

[1.10.1 相关讲解 18](#_Toc465667073)

[1.10.2 优点 18](#_Toc465667074)

[1.10.3 下载及安装 19](#_Toc465667075)

[1.10.4 参数详解 19](#_Toc465667076)

[1.10.5 基本命令 19](#_Toc465667077)

[1.10.6 事务 20](#_Toc465667078)

[1.11 内部类 20](#_Toc465667079)

[1.11.1 相关讲解 20](#_Toc465667080)

[1.11.2 类型 20](#_Toc465667081)

[1.12 注解 21](#_Toc465667082)

[1.12.1 相关讲解 21](#_Toc465667083)

[1.12.2 类型 21](#_Toc465667084)

[1.13 JQuery Validate 21](#_Toc465667085)

[1.13.1 相关讲解 21](#_Toc465667086)

[1.14 SVN 21](#_Toc465667087)

[1.14.1 相关讲解 21](#_Toc465667088)

[1.15 Log4j 22](#_Toc465667089)

[1.15.1 相关讲解 22](#_Toc465667090)

[1.16 Shiro 22](#_Toc465667091)

[1.16.1 视频教学 22](#_Toc465667092)

[1.16.2 相关讲解 22](#_Toc465667093)

[1.16.3 异常类型 22](#_Toc465667094)

[1.16.4 验证规则 22](#_Toc465667095)

[1.16.5 Shiro授权方式 23](#_Toc465667096)

[1.16.6 验证Shiro是否拥有角色方法 23](#_Toc465667097)

[1.16.7 验证Shiro是否拥有权限方法 24](#_Toc465667098)

[1.16.8 Shiro会话管理 25](#_Toc465667099)

[1.16.9 SSL 25](#_Toc465667100)

[1.17 Maven 25](#_Toc465667101)

[1.17.1 Maven搜索网址 25](#_Toc465667102)

[1.18 Junit 25](#_Toc465667103)

[1.18.1 Assert.assertEquals(Object[] expected, Object[] actual) 25](#_Toc465667104)

[1.19 Spring 25](#_Toc465667105)

[1.19.1 @ Transactional(readOnly = true) 25](#_Toc465667106)

[1.19.2 @Resource和@Autowired 25](#_Toc465667107)

[1.20 Java Web 26](#_Toc465667108)

[1.20.1 Forward和Redirect 26](#_Toc465667109)

[1.21 SSL 26](#_Toc465667110)

[1.22 Ehcache 27](#_Toc465667111)

[1.22.1 使用 27](#_Toc465667112)

[1.23 正则表达式 27](#_Toc465667113)

[1.23.1 手机号 27](#_Toc465667114)

[1.23.2 特殊字符 27](#_Toc465667115)

[1.23.3 电子邮箱 27](#_Toc465667116)

[1.23.4 IP地址 27](#_Toc465667117)

[1.23.5 日期 28](#_Toc465667118)

[2 金立日常工作笔记 29](#_Toc465667119)

[2.1 2016.09.05-2016.09.09 29](#_Toc465667120)

[2.2 2016.09.12-2016.09.18 29](#_Toc465667121)

[2.3 2016.09.19-2016.09.23 29](#_Toc465667122)

[2.4 2016.09.26-2016.09.30 30](#_Toc465667123)

[2.5 2016.10.08-2016.10.14 30](#_Toc465667124)

[2.6 2016.10.24-2016.10.28 30](#_Toc465667125)

[3 生活笔记 31](#_Toc465667126)

[3.1 好词好句 31](#_Toc465667127)

[3.1.1 励志 31](#_Toc465667128)

[3.1.2 情感 31](#_Toc465667129)

[3.1.3 生活 31](#_Toc465667130)

[3.2 灵感创意 31](#_Toc465667131)

[3.2.1 记忆旅行 31](#_Toc465667132)

[3.2.2 学校社交新型创意 32](#_Toc465667133)

[3.2.3 Jokesg：专注于广告的平台 32](#_Toc465667134)

[3.2.4 虚拟投影 32](#_Toc465667135)

[4 其它 33](#_Toc465667136)

[5 忠告 34](#_Toc465667137)

# 笔记

## Java

### JDK、JRE、JVM

JDK是 Java 语言的软件开发工具包，主要用于移动设备、嵌入式设备上的java应用程序。JDK是整个java开发的核心，它包含了JAVA的运行环境（JRE），JAVA工具和JAVA基础的类库。

Java（Java Runtime Environment）运行时环境，它包括Java虚拟机（jvm）、Java核心类库和支持文件。它不包含开发工具(JDK)--编译器、调试器和其它工具，JRE需要辅助软件--Java Plug-in--以便在浏览器中运行applet。如果你只需要运行Java程序或Applet，下载并安装它即可。如果你要自行开发 Java软件，请下载JDK(JRE和JDK的区别)，在[JDK](http://baike.baidu.com/view/25214.htm)中附带有JRE。

Java（Java Virtual Machine）虚拟机，它是一个虚构的计算机，作用就是“一次编译，多处运行”

**JVM执行程序的过程 ：**

1. 加载.class文件
2. 管理并分配内存
3. 执行垃圾收集

### Instanceof

它是一个二元操作符（运算符），和 == < >是同一类。作用是判断左边的对象是否是右边类的实例

### keytool

JDK命令：用来生成证书,包含证书、公钥及密钥。可用命令生成证书到D:\localhost.keystore：

keytool -genkey -keystore "D:\localhost.keystore" -alias localhost -keyalg RSA

自定义证书

口令:jokesg

名字和姓氏: stealth.Lee

组织单位名称:gg.com

组织名称: jokesg.com

城市区域名称:bijing

省/市/自治区名称:beijing

双字母国家/地区代码:cn

CN=stealth.Lee, OU=gg.com, O=jokesg.com, L=beijing, ST=beijing, C=cn

<localhost> 的密钥口令: jokesg

### 序列化和反序列化

序列化是将对象状态转换为可保持或传输的格式的过程。与序列化相对的是反序列化，它将流转换为对象。这两个过程结合起来，可以轻松地存储和传输数据。

主要用于存储对象状态为另一种通用格式，比如存储为二进制、xml、json等等，把对象转换成这种格式就叫序列化，而反序列化通常是从这种格式转换回来。

使用序列化主要是因为跨平台和对象存储的需求，因为网络上只允许字符串或者二进制格式，而文件需要使用二进制流格式，如果想把一个内存中的对象存储下来就必须使用序列化转换为xml（字符串）、json（字符串）或二进制（流）

**序列化：**

## 将项目部署在linux服务器

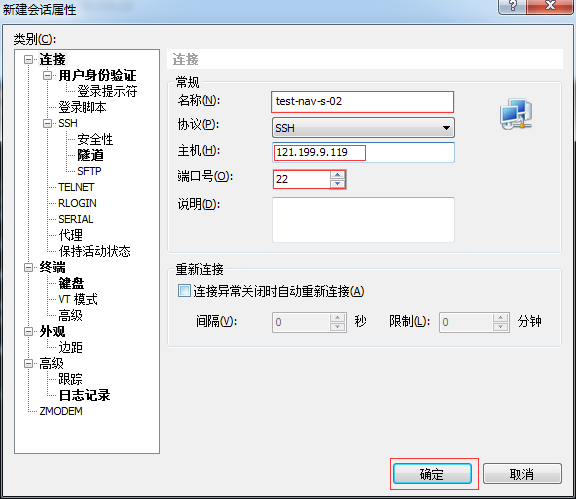
### Linux命令大全

<http://www.runoob.com/linux/linux-command-manual.html>

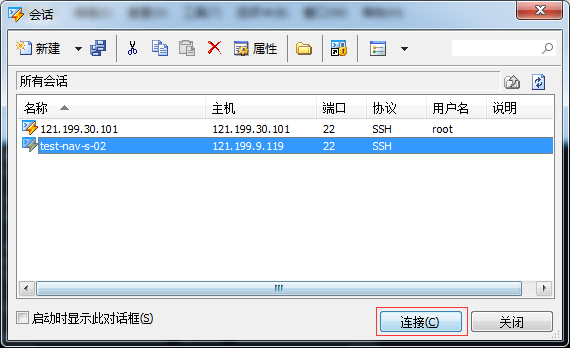
### 部署项目

1. 在Xshell 5中新建会话

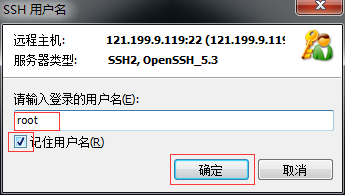
**1.**



**2.**



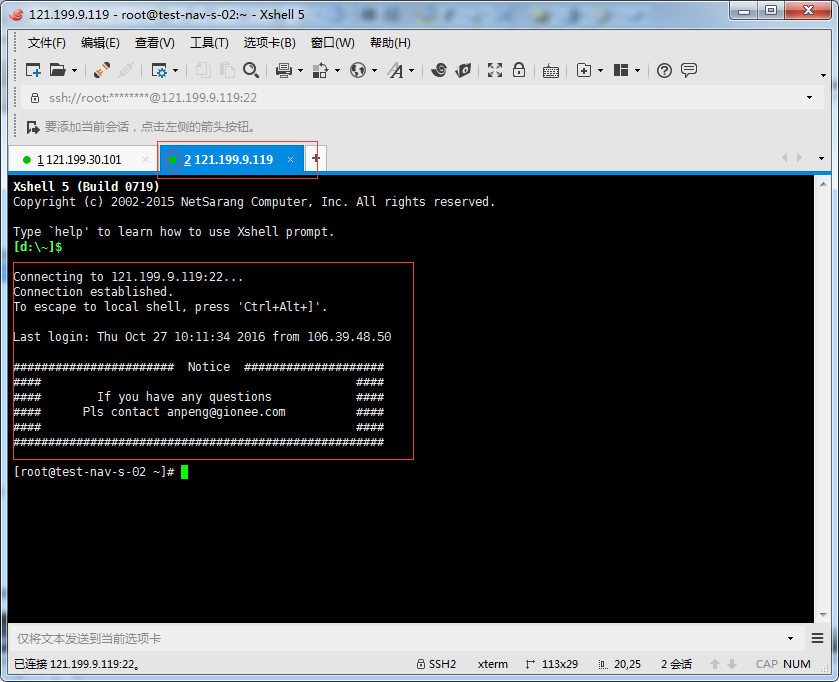
**3.**



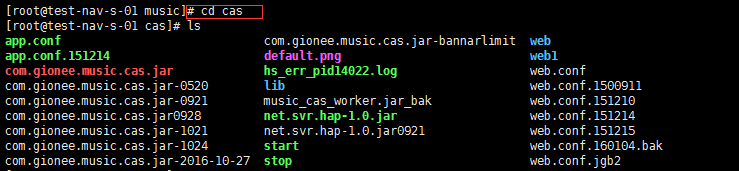
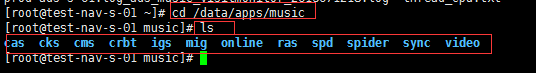
**4.**



**5.**

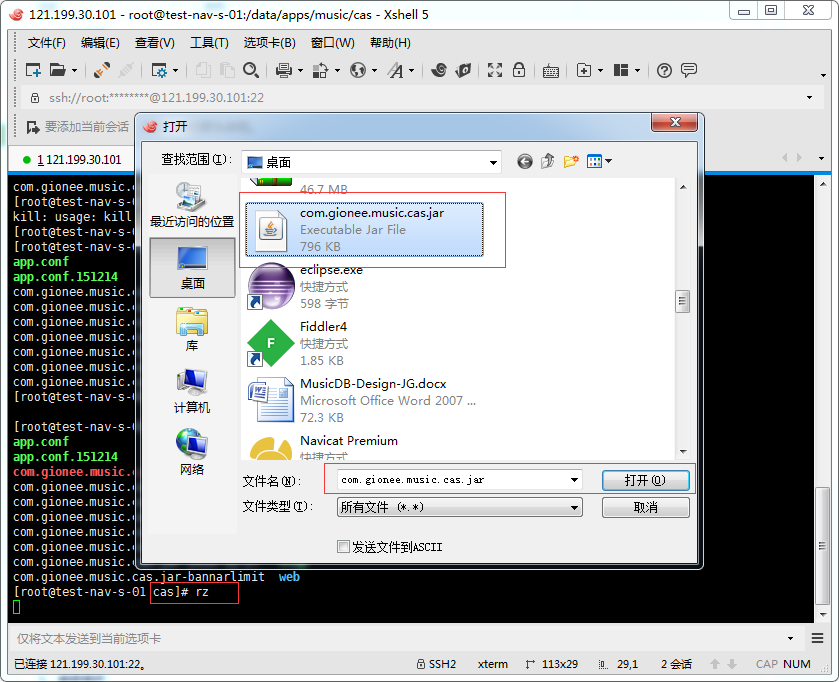


1. 将项目打成jar包（只需要main就可以，如果有依赖项目，请一并加入）
2. 进入到我们项目在服务器的目录下，例：

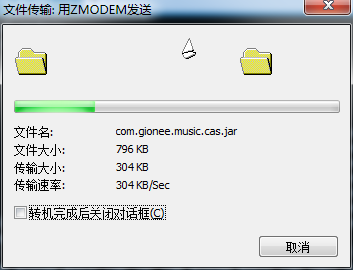


1. 上传我们刚刚打好的jar包

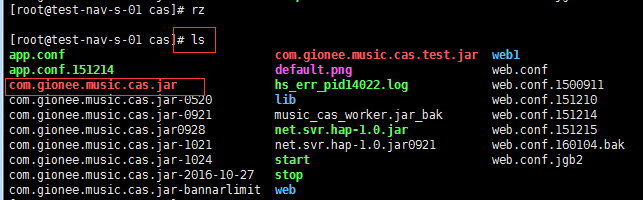
**1.**



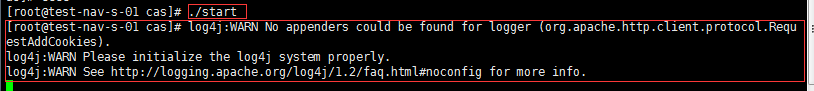
**2.**



**3.**



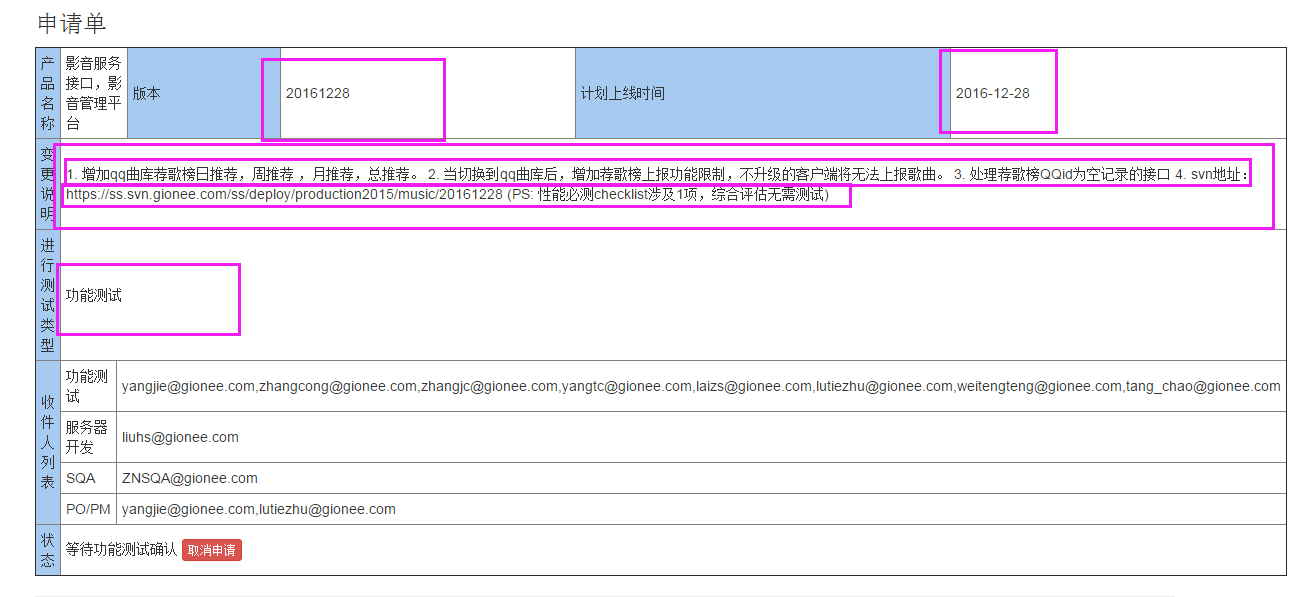
1. 启动服务器



1. 进行测试

### 金立项目上线流程

1. git pull 获取远端最新的
2. cp 文件 把自己修改的文件拷贝到目标目录中
3. git status 查看状态
4. git add . 把全部修改的文件添加到暂存区
5. git commit -m "update20161221-V1" 提交到本地仓库中，并添加简单描述
6. git push -u origin master:refs/for/master 将代码推送到远程主干上



### Fiddler4抓取数据包

1. 配置手机代理，ip:本机IP 端口：8888
2. 打开fiddler4查看有问题接口

## Scheduler:定时器

### 应用场景

自动更新数据

### 使用

创建调度工厂

1. SchedulerFactory sf = new SchedulerFactory();

根据调度工厂获取调度实例

1. Scheduler sched = sf.getScheduler();

根据JobBuilder构建Job,执行MyClass，完成目的

1. JobBuilder.newJob(MyClass.class).withIdentity(jobName,jobGroup).usingJobData(jdm).build();

通过TriggerBuilder构建Trigger，设置定时启动时间

1. TriggerBuilder.newTrigger().startNow().withIdentity(triggerName, jobGroup).withSchedule(CronScheduleBuilder.cronSchedule(cron)).build();

组装组件<job,trigger>

1. sched.schedulerJob(job,trigger);
2. startsched.start();

## 数据库

### 删除数据

**问题**：当一张表中的数据太多。1个亿。我们想要清空这张表的解决方案。用delete from table 会比较麻烦，因为数据太多可能造成删除太慢，或删除失败。

**解决方案a)：**

使用truncate table tablename

该方案效率较高。但是容易造成数据丢弃

**解决方案b)：**

1.rename table1 2.create table 3.drop table1

该方案可以写成事务，相比truncate造成数据丢弃较少

### 数据库索引

它是数据库管理系统中一个排序的数据结构，作用是为了加快数据查询，它是一个`B树`或是`B+树`

**类型**：

* 普通索引(Normal)
* 唯一索引(Unique)
* 全文索引(Full Text)
* 主键索引

**优点**：

* + - 创建索引可以大大的提高系统的性能。
    - 加快检索速度，节省查询时间，提高效率（创建索引的主要原因）
    - 创建唯一索引，可以保证数据库中数据的唯一性

**缺点**：

* 创建和维护索引需要消耗时间，这种时间随数据的增加而增加
* 索引需要占用一定的物理空间
* 当表中的数据进行增加、修改、删除的同时，索引也要动态的维护，降低了数据的维护速度

一般来说，有以下几种情况的列上不适合建立索引

* 对那些定义text/image/bit数据类型的列，即数据太大或数据太小
* 不经常用的表，数据很少的表
* 检索数据需要全文检索的时候
* 当修改性能 > 检索性能的时候

### 数据库引擎

它是用与存储、处理、和保护数据的核心服务，数据库的所有操作都是有它完成的,目的是控制访问权限并快速处理事务，从而满足企业处理大量数据的应用程序的要求

**类型：**Mysql在缺省(默认)的情况下支持三个引擎：

* ISAM
* MYISAM(默认)
* HEAP

另外两种类型：INNODB、BERKLEYDB(BDB).

## SSM

### 搭建环境

1. 添加依赖
2. 创建jdbc.propreties
3. 创建mybatis-config.xml
4. 创建spring-mvc.xml
5. 创建spring-mybatis.xml
6. 修改web.xml
7. 部署tomcat服务器

## 反射

### 相关讲解

* [**http://blog.csdn.net/cookieweb/article/details/7056277**](http://blog.csdn.net/cookieweb/article/details/7056277)

### 运用

三种方式得到Class对象

1. 对象.getClass(); 返回Class对象
2. Class.forName(“类的名称”) ；
3. Class clazz = 类名.Class

## Dom4j解析xml文件

导包：dom4j-1.6.1.jar.

想要解析一个xml文件。必须要获取Document对象。

获取Document对象的方式

SAXReader reader = new SAXReader();

Document document = reader.read(new File(“pathName”));

Element root= document.getRootElement(); 获取根节点

Root.elements();获取该节点下的所有节点

Root.element(“a”) ; 获取该节点下名为a的子节点

element.attribute("id")获取该节点ID属性

## 高并发

### 耦合性

也称块间联系。指软件系统结构中各模块间相互联系紧密程度的一种度量。模块之间联系越紧密，其耦合性就越强，模块的独立性则越差。模块间耦合高低取决于模块间接口的复杂性、调用的方式及传递的信息

### 内聚性

又称块内联系。指模块的功能强度的度量，即一个模块内部各个元素彼此结合的紧密程度的度量。若一个模块内各元素（语名之间、程序段之间）联系的越紧密，则它的内聚性就越高。

## Mysql

### 内连接

Table A Table B

ID NAME ID NAME

1 tom 1 job

2 xiaoMing 2 Tom

3 xiaoHong 3 skies

4 xiaoFang 4 xiaoHong

**Inner join**

SELECT \* FROM A INNER JOIN B ON A.NAME = B.NAME

结果是它们的交集

ID NAME ID NAME

1 tom 2 Tom

3 xiaoHong 4 xiaoHong

SELECT \* FROM A FULL OUTER JOIN B ON A.NAME = B.NAME

结果是它们的并集，注意，对于没有匹配的记录以NULL补

SELECT \* FROM A LEFT OUTER JOIN B ON A.NAME = B.NAME

结果是表A的完全集，B中匹配的则有值，不匹配NULL补

**相关讲解：**

* <http://wenku.baidu.com/link?url=ewPnBsCVyDNPG0vJFZHHbNyLMXnKw1eM9-CFoIzY9qS4adhUCjar8DfTB1AP0qjHBia2cHoYGXlAFTPi-v6o9cxfuBFTzdok0Up4Djmiv33>

### SELECT LAST\_INSERT\_ID()

自动获取最后insert或update的最后一条ID

### DATE\_SUB()

函数从日期减去指定的时间间隔。

**相关讲解：**<http://www.w3school.com.cn/sql/func_date_sub.asp>

### Explain

Explain 检索sql语句查询效率

**相关讲解：** <http://blog.csdn.net/mengxianhua/article/details/8771993>

### ****SELECT IFNULL(expr1,expr2)****

**如果expr1不为空返回expr1否则返回expr2**

## ****Redis****

Requirepass(密码)：stealth

Redis是一个开源的使用ANSI C语言编写、遵守BSD协议、支持网络、可基于内存亦可持久化的日志型、Key-Value数据库，并提供多种语言的API。它通常被称为数据结构服务器，因为值（value）可以是 字符串(String), 哈希(Map), 列表(list), 集合(sets) 和 有序集合(sorted sets)等类型。

### 相关讲解

* [http://www.redis.net.cn](http://www.redis.net.cn/tutorial/3525.html)
* <http://www.yiibai.com/redis/redis_quick_guide.html>

### 优点

* + Redis支持数据的持久化，可以将内存中的数据保持在磁盘中，重启的时候可以再次加载进行使用。
  + Redis不仅仅支持简单的key-value类型的数据，同时还提供list，set，zset，hash等数据结构的存储。
  + Redis支持数据的备份，即master-slave模式的数据备份
* 性能极高 – Redis能读的速度是110000次/s,写的速度是81000次/s 。
* 丰富的数据类型 – Redis支持二进制案例的 Strings, Lists, Hashes, Sets 及 Ordered Sets 数据类型操作。
* 原子 – Redis的所有操作都是原子性的，同时Redis还支持对几个操作全并后的原子性执行。
* 丰富的特性 – Redis还支持 publish/subscribe, 通知, key 过期等等特性。

### 下载及安装

**下载地址：**<https://github.com/dmajkic/redis/downloads>

**运行：**使用cd命令切换目录到 C:\redis 运行 **redis-server.exe redis.conf 。打开新的CMD再次运行redis-cli.exe -h 127.0.0.1 -p 6379**

**设置键值对 ：et myKey abc**

**取出键值对 ：get myKey**

**设置HASH表：HMSET user:1 username password name age**

**取出HASH表：HGETALL user:1**

**设置LIST ：**lpush name zs

lpush name ls

**取出LIST ：**lrange name 0 10

### ****参数详解****

<http://www.redis.net.cn/tutorial/3504.html>

### 基本命令

命令详解：<http://www.redis.net.cn/order/>

PING：检测redis服务是否启动

DEL：删除键

SAVE：用于创建当前数据库的备份

BGSAVE:创建reids备份文件也可以用bgsave，该命令在后台执行

CONFIG GET requirepass:查看是否设置了密码验证

CONFIG SET requirepass:设置密码验证

AUTH:认证

### 事务

事务详解：<http://www.redis.net.cn/tutorial/3515.html>

**MULT**一个事务， **EXEC** 命令触发事务**I** 开始

## Ehcache

EhCache 是一个纯Java的进程内缓存框架，具有快速、精干等特点，是一种广泛使用的开源Java分布式缓存。主要面向通用缓存,Java EE和轻量级容器。它具有内存和磁盘存储，缓存加载器,缓存扩展,缓存异常处理程序,一个gzip缓存servlet过滤器,支持REST和SOAP api等特点。

### 相关讲解

<http://www.cnblogs.com/hoojo/archive/2012/07/12/2587556.html>

### 使用

1. 导包ehcache-core-2.5.2.jar，ehcache-web-2.0.4.jar
2. 添加配置文件添加配置文件ehcache.xml、ehcache.xsd

## 内部类

可以将一个类定义在另一个类里面或者方法里面

### 相关讲解

* <http://www.cnblogs.com/dolphin0520/p/3811445.html>

### 类型

成员内部类、局部内部类、匿名内部类和静态内部类

* + **成员内部类**：最普通的内部类，它的定义位于另一个类的内部。成员内部类可以无条件的访问外部类的方法和属性。外部类访问内部类的属性和方法时不需NEW一个内部类的对象

## 注解

### 相关讲解

* + <http://www.cnblogs.com/peida/archive/2013/04/24/3036689.html>

### 类型

标记注解，一般注解，元注解

1. **标记注解：**

* @Override:重写
* @Deprecated:旧
* @SuppressWarnings:压制警告

1. **一般注解：**

* @Generated:标识计算机产生的源代码

1. **元注解：**

* @Inherited:标记继承关系
* @Documented:标记生成javadoc
* @Retention:注解的生存期
* @Target:标注的目标

## JQuery Validate

### 相关讲解

* <http://www.runoob.com/jquery/jquery-plugin-validate.html>

## SVN

### 相关讲解

* <http://www.cnblogs.com/xiaobaihome/archive/2012/03/20/2407610.html>
* <http://www.cnblogs.com/xiaobaihome/archive/2012/03/20/2407979.html>

## Git

### 相关讲解

<http://www.liaoxuefeng.com/wiki/0013739516305929606dd18361248578c67b8067c8c017b000/00137396287703354d8c6c01c904c7d9ff056ae23da865a000>

### 安装（Windows下）

以下为Windows操作系统下的安装教程，其它操作系统详见相关讲解 – 安装

1. 下载镜像：[https://git-for-windows.github.io](https://git-for-windows.github.io/)完成后，默认安装即可。
2. 打开“Git Bash”， 弹出命令窗说明，安装成功
3. 安装完成后，需要进一步设置

$ git config --global user.name "Your Name"

$ git config --global user.email "email@example.com"

### 创建版本库

版本库又名仓库，英文名Repository.这个目录里面所有文件都可以被git管理起来，每个文件修改删除都能跟踪，以便任何时刻都可以追踪历史

1. 在任意盘下新建一个空文件夹。例：workspace
2. 进入到workspace文件下，右键选择Git Bash Here，进入命令窗口。如下图
3. 在命令窗口输入“git init”。 workspace文件夹下生成.git文件，说明创建版本库成功

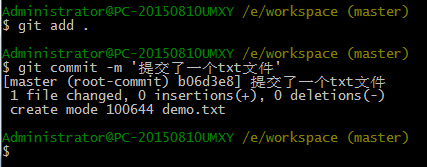
$ git init

1. 接下来就可以在版本库中添加文件 。如下图

$ git status // 此命令是查看文件状态

$ git add . // 这个命令是把所有修改过的文件添加到暂存区

$ git commit –m ‘这里是注释’// 此命令为把代码提交到本地仓库



### 常用命令

$ git init // 初始化git仓库

$ git status // 此命令是查看文件状态

$ git add . // 这个命令是把所有修改过的文件添加到暂存区

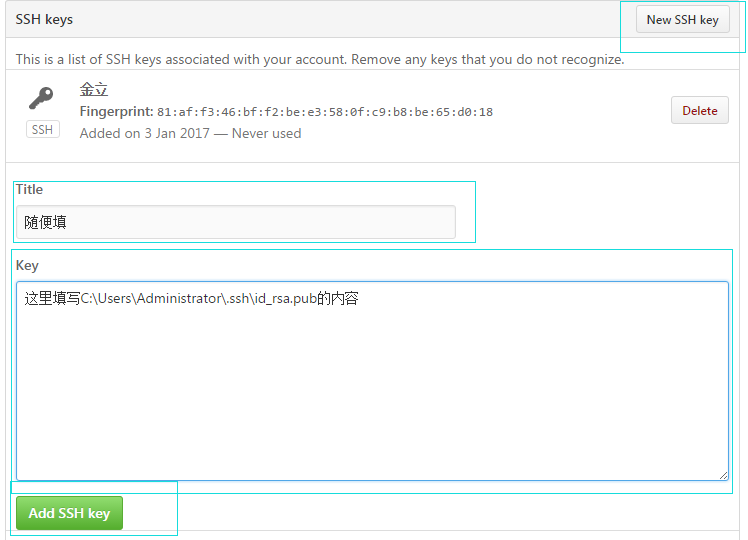
$ git commit –m ‘这里是注释’// 此命令为把代码提交到本地仓库

$ git push -u origin // 把本地仓库push到远端

$ git diff 文件名 // 对比文件

### 将代码推送到主干

1. 创建一个GitHub账号，网址<https://github.com/>
2. 首先生成SSH Key，生成方式如下
3. 打开Git Bash，输入命令 $ ssh-keygen -t rsa -C [youremail@example.com](mailto:youremail@example.com)
4. 进入C:\Users\Administrator\.ssh看这个目录下有没有id\_rsa和id\_rsa.pub这两个文件
5. 登录GitHub。打开Settings，选择SSH and GPG keys ，New SSH key。



1. 在GitHub上创建一个新的仓库。选择`Create a new repo`按提示创建即可
2. 创建完成之后关联我们的本地仓库。如下
3. 在我们的工作区目录下打开git命令窗口，输入以下内容
4. git remote add origin <https://github.com/stealth-Lee/sbuzhi.git>
5. git push –u origin master

## Log4j

### 相关讲解

* <http://www.codeceo.com/article/log4j-usage.html>

## Shiro

### ****视频教学****

* + <http://www.java1234.com/a/yuanchuang/shiro/2015/1104/5181.html>;
  + <http://www.konghao.org/videos/40>

### ****相关讲解****

* <http://jinnianshilongnian.iteye.com/blog/2018398>
* <https://my.oschina.net/u/1448456/blog/198648>

### 异常类型

* + DisabledAccountException （禁用的帐号）
  + LockedAccountException （锁定的帐号）
  + UnknownAccountException（错误的帐号）
  + ExcessiveAttemptsException（登录失败次数过多）
  + IncorrectCredentialsException （错误的凭证）
  + ExpiredCredentialsException （过期的凭证）

### 验证规则

* FirstSuccessfulStrategy：只要一个Realm验证成功即可，只返回第一个身份验证成功的Realm的认证信息，其它的忽略
* AtLeastOneSuccessfulStrategy：只要有一个Realm验证成功即可，并返回所有Realm身份验证成功的认证信息；
* AllSuccessfulStrategy**：**所有Realm验证成功才算成功，且返回所有Realm身份验证成功的认证信息，如果有一个失败就失败了。

ModularRealmAuthenticator**默认使用**AtLeastOneSuccessfulStrategy策略。

### Shiro授权方式

Shrio授权方式支持三种方式

1. **Java代码** 
   1. Subject subject = SecurityUtils.getSubject();
   2. **if**(subject.hasRole(“admin”)) {
   3. //有权限
   4. } **else** {
   5. //无权限
   6. }
2. **注解**
   1. @RequiresRoles("admin")
   2. **public** **void** hello() {
   3. //有权限
   4. }
3. **Jsp标签**
   1. <shiro:hasRole name="admin">
   2. <!— 有权限 —>
   3. </shiro:hasRole>

### 验证Shiro是否拥有角色方法

1. **判断是否拥有角色，返回boolean：**

* **boolean Subject.hasRole(String roleIdentifier)：**判断是否拥有角色roleIdentifier，拥有返回true
* **boolean Subject.hasAllRoles(Collection<String> roleIdentifiers)：**判断是否拥有角色链roleIdentifiers，全部拥有返回true
* **boolean[] Subject.hasRoles(List<String> roleIdentifiers)：一一**判断是否拥有角色链roleIdentifiers，返回boolean数组

1. **判断是否拥有角色，没有权限抛出异常**[**AuthorizationException**](eclipse-javadoc:%E2%98%82=jokesg/C:%5C/Users%5C/Administrator%5C/.m2%5C/repository%5C/org%5C/apache%5C/shiro%5C/shiro-core%5C/1.2.3%5C/shiro-core-1.2.3.jar%3Corg.apache.shiro.subject(Subject.class%E2%98%83Subject~checkRoles~Ljava.util.Collection%5C%3CLjava.lang.String;%3E;%E2%98%82AuthorizationException)**：**

* **void** [**Subject**](eclipse-javadoc:%E2%98%82=jokesg/C:%5C/Users%5C/Administrator%5C/.m2%5C/repository%5C/org%5C/apache%5C/shiro%5C/shiro-core%5C/1.2.3%5C/shiro-core-1.2.3.jar%3Corg.apache.shiro.subject(Subject.class%E2%98%83Subject)**.checkRoles(**[**Collection**](eclipse-javadoc:%E2%98%82=jokesg/C:%5C/Users%5C/Administrator%5C/.m2%5C/repository%5C/org%5C/apache%5C/shiro%5C/shiro-core%5C/1.2.3%5C/shiro-core-1.2.3.jar%3Corg.apache.shiro.subject(Subject.class%E2%98%83Subject~checkRoles~Ljava.util.Collection%5C%3CLjava.lang.String;%3E;%E2%98%82Collection)**<**[**String**](eclipse-javadoc:%E2%98%82=jokesg/C:%5C/Users%5C/Administrator%5C/.m2%5C/repository%5C/org%5C/apache%5C/shiro%5C/shiro-core%5C/1.2.3%5C/shiro-core-1.2.3.jar%3Corg.apache.shiro.subject(Subject.class%E2%98%83Subject~checkRoles~Ljava.util.Collection%5C%3CLjava.lang.String;%3E;%E2%98%82String)**> roleIdentifiers) throws** [**AuthorizationException**](eclipse-javadoc:%E2%98%82=jokesg/C:%5C/Users%5C/Administrator%5C/.m2%5C/repository%5C/org%5C/apache%5C/shiro%5C/shiro-core%5C/1.2.3%5C/shiro-core-1.2.3.jar%3Corg.apache.shiro.subject(Subject.class%E2%98%83Subject~checkRoles~Ljava.util.Collection%5C%3CLjava.lang.String;%3E;%E2%98%82AuthorizationException)**：**判断是否拥有角色链roleIdentifiers，没有权限抛出异常
* **void Subject.checkRole(String roleIdentifier) throws AuthorizationException：**判断是否拥有角色roleIdentifier，没有权限抛出异常

### 验证Shiro是否拥有权限方法

1. **判断是否拥有权限，返回boolean**

* **boolean** [**Subject**](eclipse-javadoc:%E2%98%82=jokesg/C:%5C/Users%5C/Administrator%5C/.m2%5C/repository%5C/org%5C/apache%5C/shiro%5C/shiro-core%5C/1.2.3%5C/shiro-core-1.2.3.jar%3Corg.apache.shiro.subject(Subject.class%E2%98%83Subject)**.isPermitted(**[**String**](eclipse-javadoc:%E2%98%82=jokesg/C:%5C/Users%5C/Administrator%5C/.m2%5C/repository%5C/org%5C/apache%5C/shiro%5C/shiro-core%5C/1.2.3%5C/shiro-core-1.2.3.jar%3Corg.apache.shiro.subject(Subject.class%E2%98%83Subject~isPermitted~Ljava.lang.String;%E2%98%82String) **permission)：**判断是否拥有权限permission，拥有返回true
* **boolean** [**Subject**](eclipse-javadoc:%E2%98%82=jokesg/C:%5C/Users%5C/Administrator%5C/.m2%5C/repository%5C/org%5C/apache%5C/shiro%5C/shiro-core%5C/1.2.3%5C/shiro-core-1.2.3.jar%3Corg.apache.shiro.subject(Subject.class%E2%98%83Subject)**.isPermittedAll(**[**String**](eclipse-javadoc:%E2%98%82=jokesg/C:%5C/Users%5C/Administrator%5C/.m2%5C/repository%5C/org%5C/apache%5C/shiro%5C/shiro-core%5C/1.2.3%5C/shiro-core-1.2.3.jar%3Corg.apache.shiro.subject(Subject.class%E2%98%83Subject~isPermittedAll~%5C%E2%98%83Ljava.lang.String;%E2%98%82String)**... permissions):**判断是否拥有权限链permissions，全部拥有返回true
* **boolean[]** [**Subject**](eclipse-javadoc:%E2%98%82=jokesg/C:%5C/Users%5C/Administrator%5C/.m2%5C/repository%5C/org%5C/apache%5C/shiro%5C/shiro-core%5C/1.2.3%5C/shiro-core-1.2.3.jar%3Corg.apache.shiro.subject(Subject.class%E2%98%83Subject)**.isPermitted(**[**String**](eclipse-javadoc:%E2%98%82=jokesg/C:%5C/Users%5C/Administrator%5C/.m2%5C/repository%5C/org%5C/apache%5C/shiro%5C/shiro-core%5C/1.2.3%5C/shiro-core-1.2.3.jar%3Corg.apache.shiro.subject(Subject.class%E2%98%83Subject~isPermitted~%5C%E2%98%83Ljava.lang.String;%E2%98%82String)**... permissions):**一一判断是否拥有权限链permissions，返回boolean数组

1. **判断是否拥有权限，不拥有抛出异常**

* **void** [**Subject**](eclipse-javadoc:%E2%98%82=jokesg/C:%5C/Users%5C/Administrator%5C/.m2%5C/repository%5C/org%5C/apache%5C/shiro%5C/shiro-core%5C/1.2.3%5C/shiro-core-1.2.3.jar%3Corg.apache.shiro.subject(Subject.class%E2%98%83Subject)**.checkPermission(**[**String**](eclipse-javadoc:%E2%98%82=jokesg/C:%5C/Users%5C/Administrator%5C/.m2%5C/repository%5C/org%5C/apache%5C/shiro%5C/shiro-core%5C/1.2.3%5C/shiro-core-1.2.3.jar%3Corg.apache.shiro.subject(Subject.class%E2%98%83Subject~checkPermission~Ljava.lang.String;%E2%98%82String) **permission) throws** [**AuthorizationException**](eclipse-javadoc:%E2%98%82=jokesg/C:%5C/Users%5C/Administrator%5C/.m2%5C/repository%5C/org%5C/apache%5C/shiro%5C/shiro-core%5C/1.2.3%5C/shiro-core-1.2.3.jar%3Corg.apache.shiro.subject(Subject.class%E2%98%83Subject~checkPermission~Ljava.lang.String;%E2%98%82AuthorizationException)：判断是否拥有权限，不拥有抛出异常
* **void** [**Subject**](eclipse-javadoc:%E2%98%82=jokesg/C:%5C/Users%5C/Administrator%5C/.m2%5C/repository%5C/org%5C/apache%5C/shiro%5C/shiro-core%5C/1.2.3%5C/shiro-core-1.2.3.jar%3Corg.apache.shiro.subject(Subject.class%E2%98%83Subject)**.checkPermissions(**[**String**](eclipse-javadoc:%E2%98%82=jokesg/C:%5C/Users%5C/Administrator%5C/.m2%5C/repository%5C/org%5C/apache%5C/shiro%5C/shiro-core%5C/1.2.3%5C/shiro-core-1.2.3.jar%3Corg.apache.shiro.subject(Subject.class%E2%98%83Subject~checkPermissions~%5C%E2%98%83Ljava.lang.String;%E2%98%82String)**... permissions) throws** [**AuthorizationException**](eclipse-javadoc:%E2%98%82=jokesg/C:%5C/Users%5C/Administrator%5C/.m2%5C/repository%5C/org%5C/apache%5C/shiro%5C/shiro-core%5C/1.2.3%5C/shiro-core-1.2.3.jar%3Corg.apache.shiro.subject(Subject.class%E2%98%83Subject~checkPermissions~%5C%E2%98%83Ljava.lang.String;%E2%98%82AuthorizationException)：判断是否拥有权限链permissions，不全部拥有抛出异常

### Shiro会话管理

1. Subject subject = SecurityUtils.getSubject();
2. Session session = subject.getSession();   //获取会话
3. session.getId();  //获取当前会话唯一标识
4. session.getHost(); //获取当前Subject主机地址

### SSL

## Maven

### Maven搜索网址

* [**http://mvnrepository.com**](http://mvnrepository.com)

## Junit

### Assert.assertEquals(Object[] expected, Object[] actual)

断言方法，**判断两个数组是否相等**

## Spring

### @ Transactional(readOnly = true)

事务注解，**开启事务只读属性**

### @Resource和@Autowired

**区别：**

* @Resource 相当于 @Autowired ，它俩都可以用来装配Bean
* @Resource 默认按照名称进行装配，而@Autowired 默认按照类型进行装配
* @Resource 是JDK1.6支持的注解，@Autowired是Spring的注解

### Spring IOC

## Java Web

### Forward和Redirect

**转发(forward):**服务器在接收到客户端的请求之后，服务器吧控制权交到另一个JSP页面手中新的JSP页面接收到请求之后根据情况是继续转交控制权或者显示页面由自己决定，到最后显示页面的整个过程就是一个页面跳转过程，在这个过程中，服务器可以把请求的数据在经过的页面进行传递，而不会担心数据的丢失

**重定向(redirect):** 服务器受到客户端请求之后，服务器发送新的链接到客户端，客户端接收到新的链接之后又重新请求收到的链接地址，在整个过程中完成之后在客户端看来是发生了一次跳转，其实是客户端请求了两次而已，所以在客户端的地址栏里网络地址自然就会改变成新的连接

**两者的区别：**

1. 请求的次数的不同，重定向总共请求了两次服务器；转发则是用户请求一次可能经过N个JSP页面由返回到用户浏览器中，是一次请求多次处理的过程；
2. 跳转过程中链接的变化，重定向在跳转中请求了两次服务器并且是两次不同的链接地址，在浏览器的地址栏可以看到两次是有变化的；转发在跳转过程中浏览器请求了一次服务器，服务器经过了n个JSP页面并没有改变请求的链接地址，因为用户只请求了一次，所以在整个跳转过程中链接地址是没有改变的，在浏览器的地址栏就可以看到
3. 目的不同，重定向只是简单的让用户访问一个新的链接，而转发是服务器要得到用户的请求内容并需要进行一部分处理的，所以两者目的之不同的

## SSL

SSL(Secure Sockets Layer 安全套接层):是为网络通信提供安全和数据完整性的一种安全协议，在传输层对网络连接进行加密

## 正则表达式

### 手机号

^1[34578]\\d{9}：^1开始为1 ，[34578]后面跟3/4/5/7/8任意一个 ，[\\d](file:///\\d){9}表示[0-9数字] 有九位

### 特殊字符

^[A-Za-z0-9]+$：英文和数字

^[\u4E00-\u9FA5]+$：中文

### 电子邮箱

^\w+([-+.]\w+)\*@\w+([-.]\w+)\*\.\w+([-.]\w+)\*$：

### IP地址

\d+\.\d+\.\d+\.\d+ ：还有不足

### 日期

^[0-9]{4}-(((0[13578]|(10|12))-(0[1-9]|[1-2][0-9]|3[0-1]))|(02-(0[1-9]|[1-2][0-9]))|((0[469]|11)-(0[1-9]|[1-2][0-9]|30)))$

## IO流

### 分类

根据处理数据类型的不同分为：字节流和字符流

区别：

1. 读写单位不同：字节流以字节（8bit）为单位，字符流以字符为单位，根据码表映射字符，一次可能读多个字节。
2. 处理对象不同：字节流能处理所有类型的数据（如图片、avi等），而字符流只能处理字符类型的数据。
3. 字节流在操作的时候本身是不会用到缓冲区的，是文件本身的直接操作的；而字符流在操作的时候下后是会用到缓冲区的，是通过缓冲区来操作文件
4. 结论：优先选用字节流。首先因为硬盘上的所有文件都是以字节的形式进行传输或者保存的，包括图片等内容。但是字符只是在内存中才会形成的，所以在开发中，字节流使用广泛。

根据流向不同分为：输入流和输出流

对输入流只能进行读操作，对输出流只能进行写操作，程序中需要根据待传输数据的不同特性而使用不同的流。

## jQuery

### 相关讲解

<http://jquery.cuishifeng.cn/>

## AngularJs

### 相关讲解

<http://www.runoob.com/angularjs/angularjs-directives.html>

### 四大核心特性

1. MVC
2. 模块化
3. 指令系统
4. 双向数据绑定

## 数据库

### 关于数据库优化的方法：

* 从业务逻辑看，有些返回数据量巨大的sql语句是否有必要，如果有必要，是否可以用limit做一下限制，一次返回的结果数不要太多
* 有些查询应该建索引没有建，根据需要建上
* 有些可以在内存缓存的，尽量缓存上，不要过于依赖数据库，因为对于大多数系统来讲，数据库的都是性能瓶颈

## 设计模式

### 单例模式：

懒汉模式和饿汉模式

1. @RequiresRoles("admin")
2. **public** **void** hello() {
3. //有权限
4. }

## 线程

线程是一个程序里面不同的执行路径

### 开启一个线程：

1：实现Rannable接口 并重写run方法

2：Thraed t = new Thraed(Rannable);

3：t.start();

或者

1：继承Thraed类重写run方法

2：Thraed.start();

最好使用实现Rannable接口开启一个线程

## 网络编程



## 从0搭建项目

本案例演示从0开始搭建一个java web项目。以Maven项目为例。开发工具Eclipse，数据库mysql,服务器tomcat/jetty

1. 创建Maven项目
2. New ----> Project ----> Maven (Maven Project) ----> 一路next(注意：选择Select An Archetype，选择org.apache.maven.archetypes maven-archetype-webapp) ----> 填写group Id、Artifact Id等信息 ----> Finish
3. 修改配置

修改web.Xml版本为3.0

修改.settings文件下org.eclipse.wst.common.project.facet.core.xml ---->

<installed facet=*"java"* version=*"1.7"*/>

<installed facet=*"jst.web"* version=*"3.0"*/>

点击项目右键选择properties ----> Resource(设置编码为UTF-8,见下图)

----> Java Build Path (修改jre 为1.7以上)

1. 添加依赖
2. 书写开发文档，数据库设计文档
3. 搭建项目

# 金立日常工作笔记

## 2016.09.05-2016.09.09

* 修改`music\_video\_banner`表结构:新增字段 Index 、StartTime 、EndTime
* 修改视频Banner增加时排序功能、新增起止时间：VideoBannerSaveHandler
* 修改视频Banner置顶功能（原置顶已经废弃）：VideoBannerOnTopHandler
* 新增视频Banner上下移动排序功能：VideoLinkOrderHandler
* 修改`music\_video\_category`表结构：新增字段 Index
* 修改频道管理排序功能 ：CategoryGetHandler
* 新增频道管理上下移动功能：CategoryLinkOrderHandler
* 新建`music\_video\_client\_version`:视频版本信息表
* 新增视频版本信息查询接口：VideoClientVersionGetHandler
* 新增视频版本信息修改增加接口：VidelClientVersionSaveHandler
* 新增视频版本信息删除接口：VideoClientVersionDeleteHandler
* 新建`music\_video\_module\_seq`：视频首页运营模块顺序管理表
* 新增视频首页模块顺序查询接口：VideoSeqGetHandler
* 新增视频首页模块顺序修改增加接口：VideoSeqSaveHandler
* 新增视频首页模块顺序删除接口：VideoSeqDeleteHandler
* 新增视频首页模块顺序上架接口：VideoSeqUpdateStatusHandler
* 修改用户昵称管理搜索接口：UserGetHandler
* 修改用户昵称管理编辑接口：UserUpdateHandler

## 2016.09.12-2016.09.18

* 完善用户管理用户评论列表接口：RecommentListHandler

1. 增加按关键字过滤信息
2. 完善按日期过滤信息

* 学习OSS

## 2016.09.19-2016.09.23

* 完善MusicDb接口：新增 getVersionOrModeltype(String versionOrModeltype)
* 完善RingListSaveHandler接口
* 完善CoverSaveHandler接口
* 完善SonglistTagAddHandler接口
* 完善SonglistSaveHandler接口
* 完善SonglistPubAuthorSaveHandler接口
* 完善YyLiveSaveHandler接口
* 以上接口修改图片上传为OSS
* 修改`music\_push\_schedule`表结构：新增`Content`,`Speed`字段
* 完善MusicLinkSaveHandler接口
* 完善MusicLinksGetHandler接口

以上接口新增两个字段。

## 2016.09.26-2016.09.30

// 完善AdParameterHandler接口

* 新增音乐视频推广小字接口：SmallFontHandler
* CAS新增音乐推广小字接口：SmallFontGet

## 2016.10.08-2016.10.14

* 完善CoverListGetHandler接口：增加状态过滤条件
* 完善CoverSaveHandler接口：修改增加时默认状态为删除状态（未上架）
* 完善CoverUpdateStatusHandler接口：修改状态

## 2016.10.24-2016.10.28

升级管理接口新增下载类型字段：Sign

* 完善MusicUpgradeGetHandler接口
* 完善MusicUpgradeSaveHandler接口

音乐Banner管理，新增生日Banner

* 完善MusicLinksGetHandler接口
* 完善MusicLinkSaveHandler接口
* 完善MusicLinkOrderHandler接口：禁止生日Banner上下移动

## 2016.10.31-2016.11.04

Push视频推送管理

* 新增MusicVideoPushSaveHandler接口：视频Push通知管理保存
* 新增MusicVideoPushHandler接口：视频Push通知管理推送和删除
* 新增MusicVideoPushGetHandler接口：视频Push通知管理获取

## 2016.11.07-2016.11.11

Csm入口net.svr.hap.VsWorker

影音大厅功能

* 新建表`music\_video\_aggregate\_home`：影音大厅
* 新增` VideoAggregateHomeDeleteHandler`接口：影音大厅删除Handler
* 新增` VideoAggregateHomeGetHandler`接口：影音大厅获取Handler
* 新增` VideoAggregateHomeMoveHandler`接口：影音大厅移动排序Handler
* 新增` VideoAggregateHomeSaveHandler`接口：影音大厅保存Handler

影音大厅logo功能

* 新建表` music\_video\_logo`：影音大厅logo
* 新增` VideoLogoDeleteHandler`接口：影音大厅logo删除Handler
* 新增` VideoLogoGetHandler`接口：影音大厅logo获取Handler
* 新增` VideoLogoSavetHandler`接口：影音大厅logo保存Handler

## 2016.11.21-2016.11.25

* 新增`IqiySearchHandler`:爱奇异搜索接口
* 新增` TencentSearchHandler`:腾讯搜索接口

## 2016.12.12-2016.12.16

* 修改表`music\_yylive`:增加字段`Source`：来源[0：yy 1：么么]
* 修改5.3直播间管理接口文档
* 完善YyLiveSaveHandler接口：直播间管理
* 完善com.gionee.music.cas.handler.music.YYLiveGet接口：增加MEME接口
* 完善com.gionee.music.cms.handler.yylive.YyLiveGetHandler接口

## 2016.12.26-2016.12.30

新增com.gionee.music.igs.handler.MusicUpdateQQIdHandler接口：处理QqId为空

新增com.gionee.music.igs.handler.MusicGetQQIdHandler接口：处理QqId为空

学习金立上线流程

## 2017.01.09-2017.01.13

完善com.gionee.video.cas.callable.aggregation.TencentSeachCallable接口：腾讯集数错乱

# 生活笔记

## 好词好句

### 励志

* The people who are crazy enough to think they can change the world are the ones who do
* 在错误的道路上后退也是一种前进
* 有志者事竟成，破釜沉舟，百二秦关终属楚，苦心人天不负，卧薪尝胆，三千越甲可吞
* 人的潜能是可以挖掘的,当你说太晚了的时候，一定要谨慎，那可能是你退却的借口.没有谁能阻止你成功，除了你自己！
* 比我优秀的人都在努力，我有什么理由放弃

### 情感

* 我走不进你的心，写不出你的梦，不管我付出多少终究是个外人.
* 君生我未生，我生君已老
* 黑暗里你跟我走，我绝不会在有光的时候甩开你的手
* 林深时见鹿,海蓝时见鲸,梦醒时见你
* 是我的不珍惜，让你爱上了别人。当我后悔想追回你的时候，一切都已经晚了。祝你幸福，我爱的人......

### 生活

* 不悔的青春.
* 生如逆旅单行道 哪有岁月可回头.

## 灵感创意

### 记忆旅行

专注于朋友圈，给未来的自己写一封信 ，多年以后，你会收到这封自己写给自己的信。

用户也可以写给其它人

2016.01

### 学校社交新型创意

专注于学校社交的网站：用户类型分学生、教师两种。

* 学生可以在APP上聊天，交友。认识同一学校的同学。还可以访问其他学校。
* 学生可以看到本班的生活记录，例如课程安排，交学费各项事
* 教师可以发布班级任务。

2016.02

### Jokesg：专注于广告的平台

专注于打广告的平台。企业将广告信息放到平台中，用极低的费用来面向用户，征集用户的创意广告。一旦征集到该用户的创意，同时用户获得该企业的征集费用

2016.08

### 虚拟投影

虚拟投影，随时随地和朋友`见面`聊天

### 加密技术

根据地址值动态加密，相同的密码，会生成不同的密文

2016.11

### 数据校验

在任何框架中都可使用

2016.11

### 结局帝

2016.11

### 视频内嵌奖励

2016.11

### 大数据分析人体结构

2016.12

### 大数据分析人体结构

2017.01

# 其它

-- 得到该用户所拥有的角色

SELECT r.Id,r.Name,r.EnName,r.InsertBy,r.InsertDate,r.UpdateBy,r.UpdateDate,r.Remarks,r.DelFlag

FROM gg\_user\_role ur , gg\_user u , gg\_role r

WHERE u.Username = '314975605@qq.com' AND u.Id = ur.UserId AND r.Id = ur.RoleId

-- 得到该用户下所有的权限

SELECT r.Name , m.\*

from gg\_user u , gg\_role r , gg\_menu m , gg\_user\_role ur ,gg\_role\_menu rm

WHERE u.Username = '314975605@qq.com' AND u.Id = ur.UserId AND r.Id = ur.RoleId

AND rm.RoleId = r.ID AND rm.MenuId = m.Id AND m.ParentId = 1

GROUP BY m.Id ORDER BY m.Index ASC

tomcat中java.lang.OutOfMemoryError: PermGen space异常处理

一、PermGen space PermGen space的全称是Permanent Generation space,是指内存的永久保存区域, 这块内存主要是被JVM存放Class和Meta信息的,Class在被Loader时就会被放到PermGen space中, 它和存放类实例(Instance)的Heap区域不同,GC(Garbage Collection)不会在主程序运行期对 PermGen space进行清理，所以如果你的应用中有很多CLASS的话,就很可能出现PermGen space错误, 这种错误常见在web服务器对JSP进行pre compile的时候。如果你的WEB APP下都用了大量的第三方jar, 其大小 超过了jvm默认的大小(4M)那么就会产生此错误信息了。

解决方法： 手动设置MaxPermSize大小 修改TOMCAT\_HOME/bin/catalina.sh 在“echo "Using CATALINA\_BASE: $CATALINA\_BASE"”上面加入以下行： JAVA\_OPTS="-server -XX:PermSize=64M -XX:MaxPermSize=128m 建议：将相同的第三方jar文件移置到tomcat/shared/lib目录下，这样可以达到减少jar 文档重复占用内存的目的。

<http://blog.sina.com.cn/s/blog_4ad042e50102ek2v.html>

<http://api.map.baidu.com/location/ip?ak=EkcGCUracXrq5VP8fzGzk1oG3DEZ3zb1&coor=bd09ll>

# 忠告

0、大学是自己的，所以别人的懒惰不是你颓废的理由，记住这点。

1、进入大学，就是一个新的环境，接触新的人，你的所有过去对于他们来说是一张白纸，这是你最好的重新塑造自己形象的时候，改掉以前的缺点，每进入一个新的环境，都应该以全新的形象出现。

2、“我是谁？”这是一个角色定位和角色认同问题。对“大学生”这一角色，如果模糊不清，就会出现角色错乱，大一、大二就会变成高四、高五；而如果认同不自信，“Ｕｎｉｖｅｒｓｉｔｙ”则成为“由你玩三年”，大学人生有可能成为“大混人生”。

3、“我来做什么”、“该怎么做”，这是个主题定位和态度问题。大学的主题是什么？是单纯求学，两耳不闻窗外事，还是修身求学，一心追求真善美？是做单一型人才还是当复合型人才？是被动求学还是主动奋斗？选择不同，最后文凭的含金量决然不同。

4、“我到哪里去”？这是个定向问题。进入大学，前途选择并没有结束，将来是进入蓝领、白领还是金领职业层次？不管怎样选择，改变命运的钥匙掌握在自己手里，成功的机遇总是偏爱有准备头脑的人。

5、经常给家里打个电话，始终记住：儿行千里母担忧。给家里亲人打电话或者写信的理由不要仅仅只是“要钱” 。

6、很多事情别人通知你了，要说谢谢，没有通知你，不要责怪，因为那些事你其实应该自己弄清楚。

7、不论男人还是女人，如果在大学里还把容貌当作重要的东西而过分重视的话，可能不会吃亏，但是早晚会吃亏。可能，很可能，也可以说是一定有可能。

8、千万别迷恋网络游戏。千万！记住我这里用的是千万！！

9、每个星期一定要抽时间出来锻炼身体，跑步、打球、爬山、短途骑行，和同学一起！

10、你可以有喝醉的时候，我们可以接受，但是你要明白和真正的朋友一醉才能让伤心事方休，否则，你只会是别人的谈资和笑柄。

11、面对不公平的东西，不要抱怨，你的不公平可能恰恰是别人的公平。所以，你不如去努力的奋斗，争取你自己最合适的公平。

12、如果你很少去图书馆的话，你就等于自己浪费了一大笔一大笔财富。所以，经常去那里，随意翻翻，都有收获。

13、男人，你长的可以不帅，但你这个人可以帅，而且，那才是真正得帅。（当然，女人也是同样得道理）所以，不必在穿着打扮上花太多心思。

14、不要因为寂寞而恋爱，不要因为无聊而恋爱，不要因为别人都在恋爱而恋爱。大学可能有真实的爱情，但是记住只是可能。很多时候他们是因为别人都谈恋爱而羡慕或者别的原因而在一起。所以，不必为任何分手而受太大的伤，记住，这里我所说的是太大的伤，真爱，还是值得追求的。

15、很多事情当你再回忆时会发现其实没什么。所以，不管你当时多么生气愤怒或者别的，都告诉自己不必这样，你会发现其实真的不必。

16、尊严是最重要的，但是在大学里，要懂得利用这个空间锻炼自己，要让自己的尊严有足够大的承受力，要知道，社会是一个最喜欢打碎人的尊严的地方，除了你自己，没人会为你保留它。

17、如果你的个性让很多人对你敬而远之，那么你的个性是失败的，个性的成功在于能吸引，而不是能排斥。

18、新学期如果你接新生的话，当被问到学校怎么样之类的问题时，你要记住你不但是这个学校的一分子，你更要给你学弟学妹带来信心，你走过大一，你应该知道那时对学长的信任多深。

19、你的确要学的有心计，但是记住，永远记住，在社会上要胜利的唯一的方法永远只是一个，那就是实力，永远不用怀疑。

20、人生百态，不要对新的看不惯的东西生气，无所谓的，比如说恋人同居问题，和我们很多人无关的。

21、学生会的主席之类的干部，如果你尊重他们，告诉你自己那是因为他们是你的学长而不是因为他们是你的上级，事实上，大家都只是学生而已，只是学生。

22、在大学里就开始训练自己的冷静力，这是一种能力的，有大事时，能安静并能快速想出办法的人，很厉害。

23、成功的方法多种多样，别不接受你看不惯的方法。

24、如果把上课不睡觉当做一种锻炼并且你做到了，那么，你很强，而且记住，其实你应该是这样的，老师再差，也比学生强，因为他们是老师。

25、永远不要嘲笑你的教师无知或者单调，因为有一天当你发现你用瞌睡来嘲弄教师实际上很愚蠢时，你在社会上已经碰了很多钉子了

26、如果周围有人嫉妒你，那么你可以把他从你的竞争者之列排除了，嫉妒人之人，难以成大事。

27、应该相信一句话：没有不可能的事情！！真的没有，只要去做，现在我们是学生，十年后呢？？二十年后呢？？

28、“我爱你“。别对很多人说这句话，在大学里，我的意思是，希望你只对一个人说，这是尊重你爱的人，更是尊重你自己的感情。

29、爱你的人，不管你接不接受，你都应该感谢对方，这是对你们的尊重。

30、在晚上，听听收音机也是种快乐和幸福。

31、别抱怨四级六级之类的东西，那是证明你能力的很好的东西。

32、QQ与MSN是联系朋友的工具，可以是交朋友的工具，但是别轻易相信QQ上的友谊，更别轻易幻想QQ上朋友得样子。更重要的是不要去见你的这样的朋友。

33、每个人都有潜在的能量，只是很容易被习惯所掩盖，被时间所迷离，被惰性所消磨。

34、你可以看不惯一些东西，但是你应该学会接受——如果你没法改变那一切的话。

35、还能冲动，表示你还对生活有激情，总是冲动，表示你还不懂生活。

36、优秀是一种习惯。希望能从大家身上看到这种习惯。

37、永远别渴望做个任何人都不得罪的人，有人反对，有人支持，然后自己做出决定才是精彩的人生。

38、在大二大三的时候就去看看学校的招聘会吧。你会感觉到一种前途的危机。同时也会给你一些前进的动力。

39、好好利用在公共场合说话的机会，展示或者锻炼，都可以。对以后会有很大的帮助。

40、别说脏话，你应该知道习惯的力量。找工作或者和别人接触时，你随便的一个字或者几个字会让你在别人心中的形象大打折扣。

41、魄力，要有魄力，即使是失败，也不要让自己的人生平庸。

42、喜欢一个人，就勇敢的告诉他或者她。追求你真正爱的人，只要自己认为值得，那就是值得的。

43、如果你的家庭一般，那么记得为了你的家人和你自己的前途，永远别乱花钱。记住，永远。如果你的家庭富足，也要永远记住你用的钱是你爸爸妈妈的血汗钱。

44、我不去想是否能够成功，既然选择了远方，便只顾风雨兼程；我不去想，身后会不会袭来寒风冷雨，既然目标是地平线，留给世界的只能是背影。

这句话对大家也很有用的，记住！在你遇到挫折的时候就想想它吧。

45、后悔是一种耗费精神的情绪。后悔是比损失更大的损失，比错误更大的错误。所以不要后悔。

46、相信时间的力量，可以冲淡很多东西。

47、永远不要瞧不起大学里的贫困生。

48、多笑笑，会慢慢让自己真的快乐起来。

49、要学会认知。认知能力的获取，是未来社会生存的一项基本技能。

50、要学会创新。当我们以全面、发展的眼光关注世界范围内的科技进步和社会发展的时候，会深深地感受到，“创新是一个民族进步的灵魂，是国家兴旺发达的不竭动力”

51、虽然雷峰是上个世纪的榜样，是过去式了，但是别忘记了，这个社会很多人都是怀旧的，如果你能做到雷峰所做的事情，那么无疑会提高你的人格魅力和存在价值。

52、朋友，你大学的朋友很可能就是你将来事业的一部分。他们会帮助你。但是你也应该让自己有帮助他们的实力，所以，你要努力，你和你的朋友会一起在将来打造一个可能很辉煌的事业。很好听是吗？但是记住，你们都要努力。

53、假期多回家看看，也许外面的世界很精彩，但是别忘记了，家里的人都在惦记着你，祖国的大好河山很壮丽，可是家里的温暖的亲情很美丽，

多利用假期回家看看，不是看父母，而是给父母看，你是永远都是他们的骄傲！

54、面试时很多老总会问你：（如果你能见到老总的话）你会做什么？也许你现在其实就可以记住这个问题，这样可能以后会好好回答（尤其对文科学生来说）

55、要学会做人。对于我们这些经过大浪淘沙走进象牙塔里的大学生来说，知识、方法可以很快掌握，但是良好的习惯，优秀的品质，高尚的品德却不是朝夕可就的。只有把求知与做人完美地结合起来，才能成长为一个对社会有用的人，才能真正实现自己的价值。

56、作为一个社会人，与社会的联系越紧密，机会也就会越多，也就越能锻炼自己，以后走上社会了，能做的事情也就越多。

57、大了，成熟了，稳重了，但是这和激情不矛盾，一种对工作和学习的冲击力及持久力会让你有特殊的魅力和个人实力。

58、从绝望中寻找希望，人生终将辉煌。

59、请不要太早学会抽烟喝酒，即使别人看起来这样很男人。如果你不抽烟的话，你的精力将比抽烟的时候好的多，这是绝对的。想要抽烟的时候，先问下周围的人可不可以，要学会尊重别人。少在女生面前耍酷抽烟，你不知道，其实她们内心很反感。

60、对陌生人，或者把对方当做一张白纸，或者把对方当你的朋友，总之别当做敌人，即使你听到再多的关于他（她）的不好的传闻。

61、大学里手机是否应该用只有你自己知道。

62、第一年不要放松，确立自己的领先位置，这样对以后的学习比较有好处。

63、计算机一定学好。对以后的继续学习很重要。千万别不当回事。

64、宿舍里不要太邋遢，否则影响效率。

65、力所能及的事情，只要我们认真做了，我们的生活就很充实；小小的愉快，我们体会到了，就是我们给自己的奖赏！

66、多看电影。提高一下艺术修养，与人交流也有的谈。

67、注意在大学里为了以后就业积累相关的工作经验。

68、参加其他系的跨学科的活动是非常有好处的。

69、多看本学科的一些杂志，掌握本学科前沿的动态。不要都大学生了还死读书。

70、大学里男生要有风度一些。女生总是有各种各样的问题需要男生来解决，这个时候男生总是充当修理工，电工，木匠的角色，要乐于助人。

71、养成良好的生活习惯。以后就没有父母唠叨你了，比如常洗澡，晚上睡觉不要老是吹电扇什么的，自己就不要放纵自己了，要不然肯定生病。

这样的教训谁都要来几次，要不然是长大不了的。

72、千万不要把贵重物品随手放在桌子上和床头。大学宿舍里不是你家，防人之心不可无。

73、要勇于学自己不会的东西。比如电脑，可能在一开始你是很厉害的，一些来自偏远地区的孩子，可能一开始什么都不知道，但是一段时间以后，由于他们见了人就请教，最后很可能成了不错的电脑高手。在大学里面一定要不耻下问，不要怕丢面子。

74、如果你有创业的念头，不妨平时多跟同学联系，或许你们可以组成一个团队来参加创业计划大赛呢。

75、大学里时间比较充裕，如果没有计划，很容易荒废过去。最好给自己一个计划吧。

76、大学教育将赋予一个自然人人格的提高，灵魂的升华；将引导一个人去认真思考怎样的一生才是有意义的人生；将重新树立你的价值观、人生观、世界观。总的说来，大学四年每个人都将接受灵魂的洗礼。

77、应该说安全始终是做任何事情的第一要素，上大学同样也不例外。爱护自己，珍惜自己，是很重要的！因为没有什么比失去健康，比失去生命更重要！请大家一定要注意安全，爱惜自己。记住：无论何时都不要自暴自弃！

78、请从今天开始学会不要轻易发火，学会控制自己的情绪是一个成年人的开始。

79、虽然电子邮件E-mail的问世给我们生活带来了很大的方便，传统的写信发方式仍然有着无法比拟的优势，正所谓“见字如面”，手捧着远方的你亲手写的信纸，那种亲切的感觉便会油然而生，朋友们，当你有空的时候不妨写封信吧。

80、作弊要靠别人，慢慢地你就会明白有时别人是靠不住的，到那时就晚了，还是相信自己，靠自己吧。做一个讲诚实，守信用的人吧！

81、“大众化时代的大学生不能再自诩为社会的精英，要怀着一个普通劳动者的心态和定位去参与就业选择和就业竞争。”

82、“态度决定一切。”不用多解释吧？

83、“近朱者赤，近墨者黑。” 大学两极分化严重，有很厉害的人物自然也就有脑死亡的笨蛋。

84、“与其诅咒黑暗，不如点亮蜡烛。”我们应该乐观地展望一下未来，并且从此刻起继续前进。

85、我们期待行动，而不是抱怨；我们应该传播希望，而不是谣言。

86、这个世界是公平的，上帝给谁的都不会太多。一个人在一些地方获取了什么，注定要在另一些地方失去些什么，这只是时间的问题。

87、永远不要为考试而学习。在大学里，成绩固然重要，但是我们应该在策略上重视它，在战略上渺视它。

88、不要束缚自己的能力。尽可能地去想一些新奇，大胆，甚至疯狂的主意，不要束缚自己的想像力，用实践去检验你的猜想。很多时候我们不是做不到，而是想不到。

89、老师会是你的朋友。老师不仅是教授你知识的人，也是能指点你前进的长者，能和你一起探讨知识的伙伴。

90、记住该记住的，忘记该忘记的。改变能改变的，接受不能改变的，用智慧去分辨两者。

91、日出东海落西山，愁也一天，喜也一天；遇事不钻牛角尖，人也舒坦，心也舒坦。

92、无知者无畏并不可怕，真正可怕的是无知者还无所谓。一定要记住：不要用无所谓的态度原谅自己、对待一切，那会使一切变得对你无所谓，也会使你成为一个无所谓而又无所成的痛苦的边缘人。

93、始终像你并不需要钱一样工作；始终像你从未受到过伤害一样恋爱；始终像没有人在注视你一样跳舞；始终，始终像你住在人间天堂一样生活。

94、世界上最美好的东西莫过于有几个头脑和心地都很正直，严正的朋友。

95、大部分女生将电脑当成了影碟机，大部分男生将电脑当成了游戏机。大学生要掌握必要的计算机操作能力，但是，很多时候电脑会成为浪费时间的堂而皇之的借口。有电脑的大学生非常多，可是，这中间很多人可能大学毕业的时候还不会Excel，不会做一个像样的PPT。

95、别以为在学生会你能得到你曾经想要的。工作的热情很可能因为你所看到的所经历的一些事而改变，不妨选择一两个自己最喜欢的社团和项目，并坚持参加。

96、刚进大学时，评价一个同学能力基本是看他的人际沟通能力，而不是分数，你要做好这样的心理准备。

97、进入学校半个月后，绝大部分同学都会经历一个对学校的失望周期，别担心，这是正常的。

98、别羡慕那么父母能提供很好条件的人，这个世界很不公平，现在你要靠自己努力去改变他。不要计较你的父母出身，这是不可改变的，你可以通过你的努力改变你和你父母的命运，尽管这很艰难。

99、学习：不是激情澎湃、动人心扉的事，更多的是细水长流、润物无声；不需要豪言壮语，只需默默努力；少有昙花一现的壮观，只有持久的痛苦前行；

可以逆水行舟，可自暴自弃；自己选择吧；

为人处事：可以斤斤计较，可以豁达；可以与人为善，可以睚眦必报；可以自我，可以分享与奉献；可以嫉妒，可以欣赏；自己选择吧；

大学生活：或成功或挫败，或自豪或抑郁，或肩负起责任或大混人生，或精彩或平淡；自己选择吧；