**Java内存模型**

Java程序运行时数据区划分如下：

程序计数器

堆

本地方法栈

虚拟机栈

方法区

1. 程序计数器：几乎不占用内存，可以看做当前线程所执行的字节码的行号指示器，通过改变程序计数器的值来选取下一条需要执行的字节码指令，是线程私有的
2. Java虚拟机栈：这也是线程私有的，生命周期与线程相同，每个方法调用时都会创建一个栈帧用于存储局部变量表，操作数栈，方法出口等信息。
3. 本地方法栈：与虚拟机栈，但是视为本地（Native）方法服务。
4. 方法区：各个线程共享区域，用于存储已被虚拟机加载的类信息，常量、静态变量。
5. Java堆划分：新生代和老年代

根据对象存活的周期不同将内存划分为几块，一般划分为老年代和新生代。GC时可以根据不同区域的特点采用不同的算法。新生代对象存活的时间比较短，老年代较长。

Java堆内存 = 新生代 + 老年代。堆内存可以使用参数-Xms、-Xmx参数指定

新生代又被划分为eden 、 from survivor、to survivor三块区域。三块区域大小默认为8:1:1。创建对象时只会使用eden和其中的一个survivor区域。即总有一块survivor内存是空闲的。当GC清理时，存活的对象年龄会+1，当对象的年龄为15(默认为15，可以使用JVM参数 MaxRnuringThreshold设置年龄)时，对象就会成为老年代。但不一定需要达到预定年龄才能进入老年代，当两个survivor内存中相同年龄的对象占了内存的一半以上时，大于等于该年龄的对象直接进入老年代。对于一些较大的对象，则直接进入老年代。

Java虚拟机判断对象是否存活算法：

1. 引用计数法：给对象中添加一个引用计数器，每当一个地方引用它时，计数器值就+1；当引用失效时，计数器值就减一。任何计数器值为0的对象都不能再被使用。这种算法实现简单，效率高，但很难解决对象之间相互循环引用的问题。
2. 可达性分析法：通过一系列的GC ROOTS对象作为起始点向下搜索，搜索走过的路径被称为引用链。当一个对象到GC ROOTS对象没有任何引用链相连，则证明该对象是不可用的，会被判定为可回收的对象

GC垃圾清除算法：

1. 标记-清除算法：对要回收的对象进行标记，标记完成后同意回收。但是标记和清除的效率不高，而且回收后会产生大量不连续的内存。如果连续内存不够，则会再次触发清理。
2. 复制算法：将内存按照容量划分为两块同样大小，每次只使用其中一块，清理时将存货的对象复制到另一块，然后再把已使用的内存一次清理掉。这种清理适合于新生代。但存货对象较多时，会降低效率
3. 标记-整理算法：步骤与标记-清理一样，但是后续步骤不是清理，而是让存活的对象向一段移动，在清理界限外的内存。适合于老年代

Mysql:一种用来存储的数据关系型数据库。

Mysql主要数据类型：

数字类型：整数类型，浮点数类型和定点数类型

字符串类型、日期和时间类型、

事务：主要用于处理操作量大，复杂度高的数据。事务必须满足4个条件：

1. 一组事务，要么成功，要么撤回
2. 稳定性：有非法数据（外键约束之类），事务撤回。
3. 隔离性，事务独立运行，如果一个事务运行完后影响了其他事务，那么其他事务会被撤回
4. 可靠性：软、硬件崩溃后，InnoDB数据表驱动会利用日志文件重构修改。可靠性和高速度不可兼得， innodb\_flush\_log\_at\_trx\_commit选项 决定什么时候吧事务保存到日志里。

使用事务：使用start transaction开始事务，savepoint 保存点名称

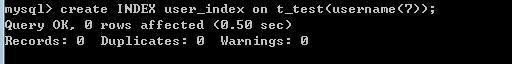
当执行出错时使用rollback回滚数据回滚到保存点的位置，如果没错就使用commit提交事务。

Mysql索引：索引可以大大提高mysql的检索速度。

索引分单列索引和组合索引。单列索引是针对表中某一行的索引。一张表可以有多个索引。组合索引是一个索引包含多个列。实际上索引也是一张表，该表保存了主键和字段，并指向实体表的记录。但是一张表过多的使用索引会照成更新标的速度，因为更新表的同时也会去更新索引。

创建索引的方式：

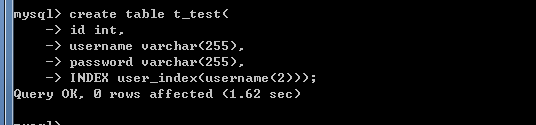
1. CREATE INDEX indexName ON TABLE\_NAME （COLUMN(LENGTH)）



1. ALTER TABLE TABLE\_NAME ADD INDEX [indexName] ON （COLUMN(LENGTH)）

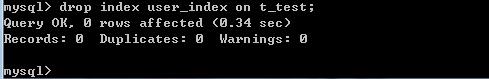


1. 创建表的时候直接指定 INDEX indexName （COLUMN(LENGTH)）

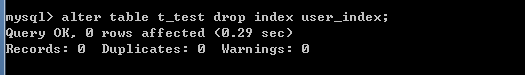


删除索引：

Delete INDEX [indexName] on TABLE\_NAME



ALTER table table\_name drop INDEX [indexName]

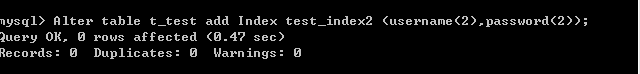


唯一索引：索引列的值必须唯一，如果是组合索引，组合值必须唯一。创建方式与普通索引类似。唯一索引使用关键字UNIQUE

全文索引：MySQL从3.23.23版开始支持全文索引和全文检索，FULLTEXT索引仅可用于 MyISAM 表；他们可以从CHAR、VARCHAR或TEXT列中作为CREATE TABLE语句的一部分被创建，或是随后使用ALTER TABLE 或CREATE INDEX被添加。

组合索引(最左前缀)：

Alter table t\_test add Index test\_index2 (username(2),password(2));



当执行下列sql语句时会使用索引：

Select \* from t\_test where username = “李xx”;

Select \* from t\_test where username = “李xx”and password = ‘123’;

但是使用这条sql语句时不会使用索引：

Select \* from t\_test where password = ‘123’;

是因为组合索引相当于建立了

Username,passwrod和username两条索引，因为最左前缀只从最左边的开始组合，所以就没有password索引。