-线程和进程区别：

线程是程序执行流的最小单位，而进程是系统进行资源分配和调度的一个独立单位。

-Thread 与Runnable、Callable区别：

Thread(start())是类，Runnable(run())和Callable是接口，Callable（call()）可以通过future返回结果。

-Lock与Synchronized区别：

1.ReentrantLock多了 锁投票，定时锁等候和中断锁等候。TryLock(time)，lockInterruptibly可以被中断的锁。

2、synchronized是在JVM层面上实现的，而且在代码执行时出现异常，JVM会自动释放锁定，但是使用Lock则不行，lock是通过代码实现的，要保证锁定一定会被释放，就必须将unLock()放到finally{}中。

3、在资源竞争不是很激烈的情况下，Synchronized的性能要优于ReetrantLock，但是在资源竞争很激烈的情况下，Synchronized的性能会下降几十倍，但是ReetrantLock的性能能维持常态

4. Group By 和 Having, Where ,Order by这些关键字是按照如下顺序进行执行的：Where, Group By, Having, Order by。

5.索引失效情况：

如果条件中有or，即使其中有条件带索引也不会使用

对于多列索引，不是使用的第一部分，则不会使用索引

like查询是以%开头

6.OOM的原因：

堆（新生代，老年代）/方法区（老年代）溢出。

堆溢出：new的对象太多。方法区溢出：加载类太多，常量静态变量太大太多。

7.内存泄露原因：

连接数据库，网络，io等连接没关闭，native方法。静态集合类 ，集合里的对象属性值被改变，单例模式持有外部对象时。

虚拟机类加载过程：

加载二进制流生成Class对象（加载）->验证符合虚拟机要求（验证）->分配空间（准备）->将常量池内符号转化为引用（解析）->初始化(clinit不是new方法)

双亲委派机制好处：

启动类加载器->扩展类加载器->应用程序类加载器(底层的加载器收到类加载请求时，首先请求父类加载器加载，父类失败在自己来)

优点：如：不然会有很多Object类。

-可达性分析的root：

被启动类加载器加载的类和创建的对象；java栈中引用的对象；方法区中静态/常量引用的对象；Native方法中JNI引用对象。

-AOP的几种实现方式：

**1.经典的基于代理的AOP：**

**2.@AspectJ注解驱动的切面**

**3.纯POJO切面**

**4.注入式AspectJ切面**

8.单例模式：

public class Singleton {//饿汉模式

private final static Singleton instance = new Singleton();

private Singleton(){}

public static Singleton getInstance(){

return s;

}

}

public enum Singleton {

INSTANCE;

public void whateverMethod() {

}

}

public class Singleton {//懒汉模式

private static volatile Singleton singleton;

private Singleton() {}

public static Singleton getInstance() {

if (singleton == null) {

synchronized (Singleton.class) {

if (singleton == null) {

singleton = new Singleton();

}

}

}

return singleton;

}

}

9. LinkedHashMap可以认为是**HashMap+LinkedList**，即它既使用HashMap操作数据结构，又使用LinkedList维护插入元素的先后顺序。

常见http状态码：

| **状态码** | **响应类别** | **原因短语** |
| --- | --- | --- |
| 1XX | 信息性状态码（Informational） | 服务器正在处理请求 |
| 2XX | 成功状态码（Success） | 请求已正常处理完毕 |
| 3XX | 重定向状态码（Redirection） | 需要进行额外操作以完成请求 |
| 4XX | 客户端错误状态码（Client Error） | 客户端原因导致服务器无法处理请求 |
| 5XX | 服务器错误状态码（Server Error） | 服务器原因导致处理请求出错 |

* 200 OK   
  请求正常处理完毕
* 204 No Content   
  请求成功处理，没有实体的主体返回
* 206 Partial Content   
  GET范围请求已成功处理
* 301 Moved Permanently   
  永久重定向，资源已永久分配新URI
* 302 Found   
  临时重定向，资源已临时分配新URI
* 303 See Other   
  临时重定向，期望使用GET定向获取
* 304 Not Modified   
  发送的附带条件请求未满足
* 307 Temporary Redirect   
  临时重定向，POST不会变成GET
* 400 Bad Request   
  请求报文语法错误或参数错误
* 401 Unauthorized   
  需要通过HTTP认证，或认证失败
* 403 Forbidden   
  请求资源被拒绝
* 404 Not Found   
  无法找到请求资源（服务器无理由拒绝）
* 500 Internal Server Error   
  服务器故障或Web应用故障
* 503 Service Unavailable   
  服务器超负载或停机维护