

堆 (Heap)

是java虚拟机所管理的内存中最大的一块内存区域，也是被各个线程共享的内存区域，该内存区域存放了对象实例及数组（但不是所有的对象实例都在堆中）

方法区 (Method Area)

它用于存储虚拟机加载的类信息、常量、静态变量、是各个线程共享的内存区域

虚拟机栈 (JVM Stack)

描述的是java方法执行的内存模型：每个方法被执行的时候都会创建一个“栈帧”，用于存储局部变量表(包括参数)、操作栈、方法出口等信息。每个方法被调用到执行完的过程，就对应着一个栈帧在虚拟机栈中从入栈到出栈的过程。声明周期与线程相同，是线程私有的。

本地方法栈

与虚拟机栈基本类似，区别在于虚拟机栈为虚拟机执行的java方法服务，而本地方法栈则是为Native方法服务

程序计数器 (PC Register)

是最小的一块内存区域，它的作用是当前线程所执行的字节码的行号指示器，在虚拟机的模型里，字节码解释器工作时就是通过改变这个计数器的值来选取下一条需要执行的字节码指令，分支、循环、异常处理、线程恢复等基础功能都需要依赖计数器完成。

直接内存

Java的NIO库允许Java程序使用直接内存，从而提高性能，通常直接内存速度回优于Java堆，读写频繁的场所会考虑使用

运行时常量池

运行时常量池是方法区的一部分，Class文件中除了有类的版本、字段、方法、接口等描述信息外，还有一项信息是常量池，用于存放编译期生成的类、方法、接口等中的常量，也包括字符串常量和符号引用，这部分内容将在类加载后进入方法区的运行时常量池中存放。

