# Java 内存区域与内存溢出异常

## 运行时数据区域

Java虚拟机在执行Java程序的过程中把内存划分成若干数据区域。



### 程序计数器

程序计数器(Program Counter Register),是一块较小的内存空间存储空间。主要存储线程下一条需要执行的字节码指令。该区域是私有的，此区域是唯一一个在Java虚拟机规范中没有规定OutOfMemmoryError情况的区域。

### Java虚拟机栈

Java虚拟机栈(Java Virtual Machine Stacks),该区域也是线程私有的。他的生命周期与线程生命周期相同。每个方法执行同时都会创建一个栈帧(Stack Frame) 用于存储局部变量表、操作数栈、动态链接、方法出入口等信息。每个方法从调用至完成的过程，就对应一个栈帧再虚拟机中，入栈道出栈的过程。

栈规定了两种异常情况，StackOverflowError:线程请求的栈深度大于虚拟机所运行的深度。OutOfMemoryError:k扩展时无法申请到足够的内存。

### 本地方法栈

本地方法栈(Native Method Stack)。和虚拟机栈的区别是，虚拟机栈执行java方法（字节）服务。本地方法栈使用到Native 方法服务。本地方法栈区域也会抛出StackOverflowError 和OutMemoryError异常。

### Java堆

Java 堆(Java Heap)。被所有线程共享，虚拟机启动时创建，存放对象实例。Java 垃圾收集器主要管理的区域。Java 堆还可以细分为 新生代和老年代，还可以细分为Eden空间、From Survivor 空间、To Survivor空间。

Java 堆可以处于在物理上不连续空间，逻辑上连续就可以可以通过-Xmx和Xms控制堆分配， 堆如果没有内存分配完成实例，并且堆现在也无法再扩展，将会抛出OutOfMemoryError异常。

### 方法区

方法区(Method Area)，多个线程共享的内存区域。存储虚拟机加载的类信息，常量、静态变量、及编译器编译后的代码等数据。Java 虚拟机规范描述方法区为堆的一个逻辑部分。

当方法区无法满足内存分配需求时，将抛出OutOfMemoryError异常。

### 运行时常量池

运行时常量池(Runtime Constant Pool)是方法区的一部分。由于存储编译期生成的各种字面量和符号引用。

当常量池无法再申请到内存时会抛出 OutOfMemoryError异常。