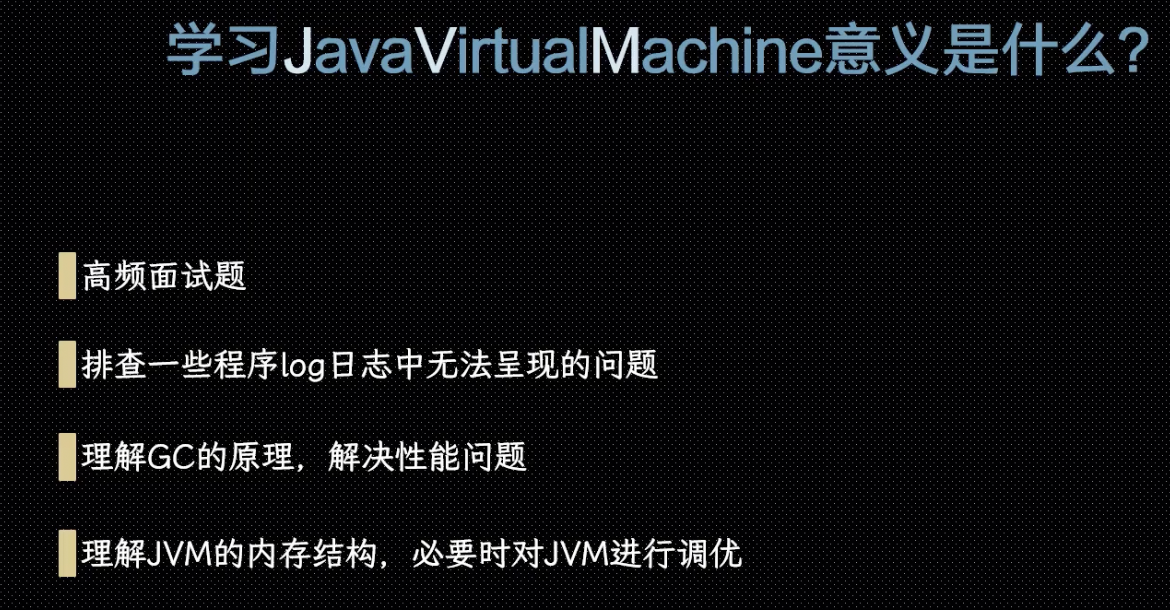
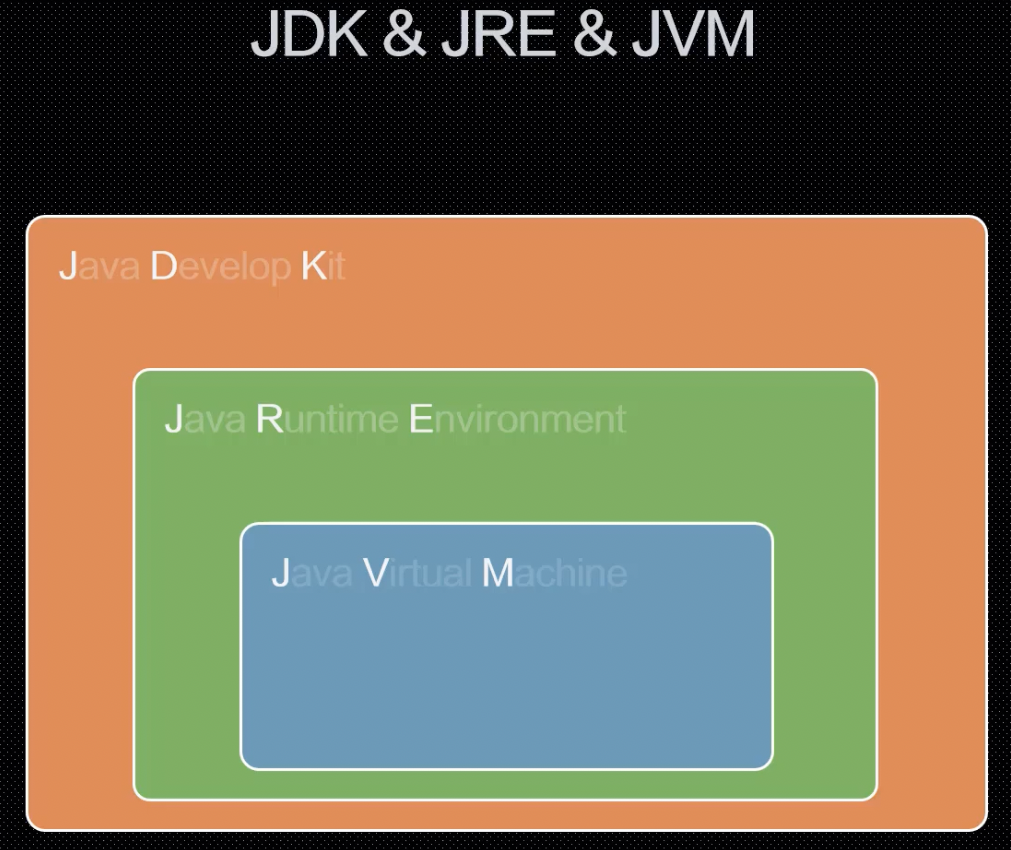
## 1.学习JVM的意义是什么？

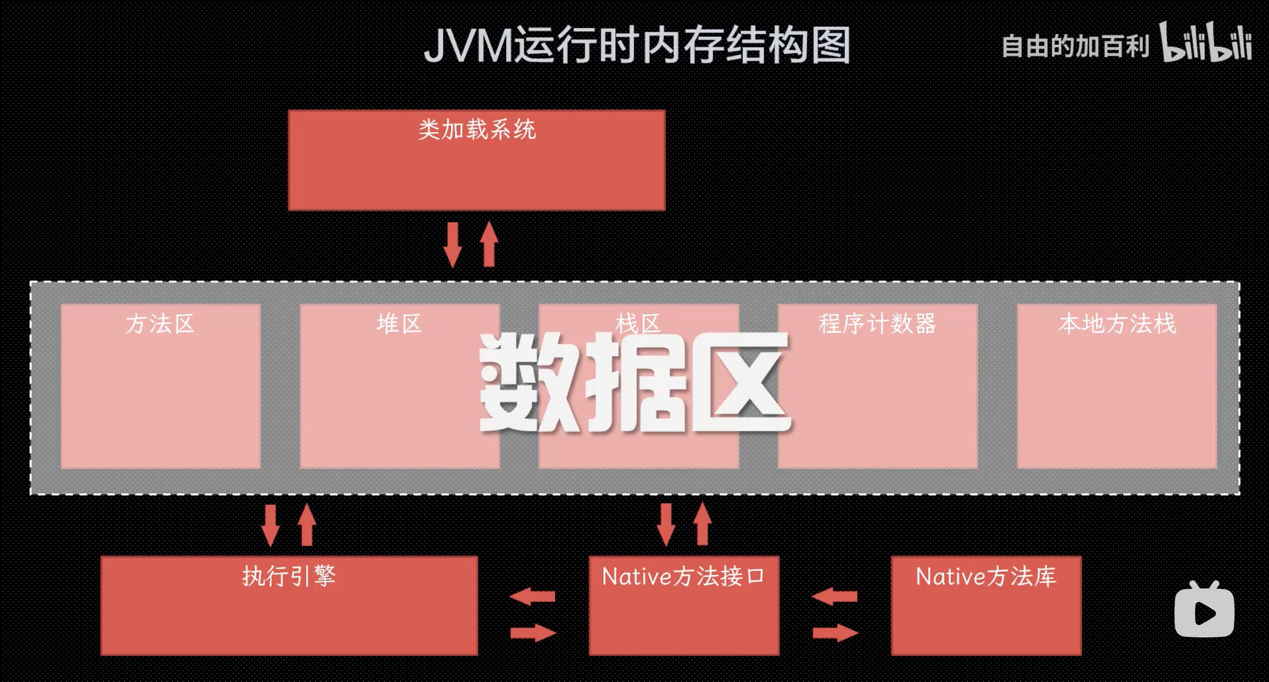


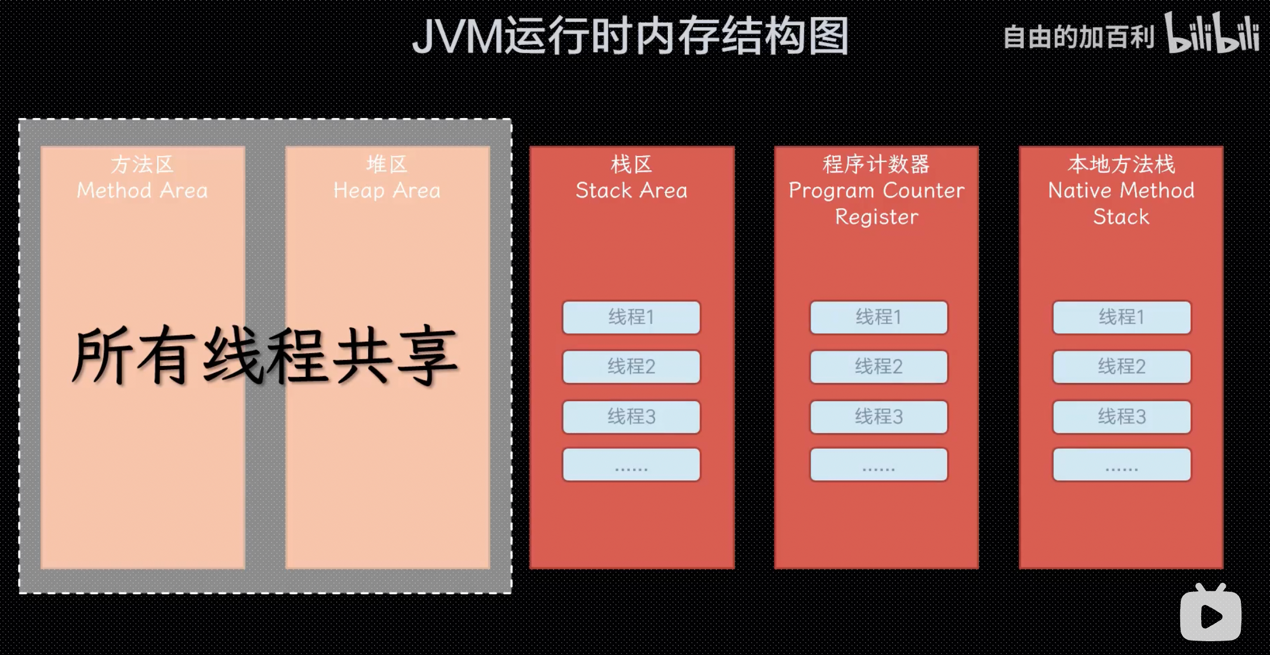
## 2.JDK&JRE&JVM的关系



* JVM（Java Virtual Machine）为java虚拟机。
* JRE包含了JVM，并且包括了运行所依赖的类库（jar包等文件）。
* JDK除了包含JRE，还含有对于开发人员需要的编译工具和反编译工具。

## 3.JVM运行时结构图





* 方法区和堆区市所有线程共享的，称为线程共享区。
* 其他区域不共享线程，称为线程独立区。

### 3.1 程序计数器

* 程序计数器是一小块内存区域，可以看作当前线程执行字节码的行号指示器，字节码解释工作是通过改变这个计数器的值来执行操作。
* 每条线程都需要自己独立的程序计数器，多条线程计数器之间不互相影响，独立存储。我们称这类内存区域为**线程私有的**内存区域。
* 此内存区域是Java虚拟机中唯一一个没有规定任何OutOfMemoryError的内存区域。

### 3.2 Java虚拟机栈（栈区、栈内存）

* Java虚拟机栈也是线程私有的，生命周期与线程相同。每个方法在在执行的时候都会创建一个栈帧。
* 栈帧中包含了局部变量表和操作数栈。
* Java虚拟机栈是用来执行Java方法的。而本地方法栈是用来执行Java内置方法，这个方法可能是C语言实现的等等。
* 在程序的执行过程避免不了方法调用方法，也就不可避免的出现Java虚拟机栈和本地方法栈之间的通信（通过JNI本地方法接口）。
* 如果线程在栈中申请的深度大于虚拟机所允许的深度，将出现StackOverFlowError异常;如果扩展无法申请到足够的内存，​ 就会抛出OutOfMemoryError异常。

### 3.3 方法区

* 方法区域Java堆都是线程共享区。
* 用于存储类的修饰符、权限、继承关系、接口列表、字段信息，声明的顺序、修饰符、类型、名字，方法的修饰符、返回值、名字、参数列表、抛出的异常、方法字节码指令、静态变量。
* 常量池存在方法区中。
* 方法去无法满足内存分配时，将抛出OutOfMemoryError异常。

### 3.4 常量池

* 用于存储常量。
* 用于存储字面量，比如“String s = “abc””，其中的“abc”为字面量。