

有关Java线程机制的浅析Java认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/594/2021_2022__E6_9C_89_E5_85_B3Java_c104_594674.htm 一 线程的基本概念：线程是一个程序内部的顺序控制流，一个进程相当于一个任务，一个线程相当于一个任务中的一条执行路径。多进程：在操作系统中能同时运行多个任务（程序）；多线程：在同一个应用程序中有多个顺序流同时执行；Java线程是通过java.lang.Thread类来实现的；VM启动时会会有一个由主方法（public static void main（）{}）所定义的线程；以通过创建Thread的实例来创建新的线程每个线程都是通过某个特定Thread对象所对应的方法run（）来完成其操作的，方法run（）称为线程体通过调用Thread类的start（）方法来启动一个线程 二 Java线程的创建和启动：可以有两种方式创建新的线程：第一种：1.定义线程类实现Runnable接口 2.Thread myThread = new Thread（target）；//target为Runnable接口类型 3.Runnable中只有一个方法：public void run（）；用以定义线程运行体 4.使用Runnable接口可以为多个线程提供共享的数据 5.在实现Runnable接口的类的run（）方法定义中可以使用Thread的静态方法public static Thread currentThread（）；获取当前线程的引用 第二种：1.可以定义一个Thread的子类并重写其run方法如：class MyThread extends Thread { public void run() {...} } 2.然后生成该类的对象：MyThread myThread = new MyThread（）； 三 Java线程控制的基本方法：isAlive（）：判断线程是否还"活"着 getPriority（）：获得线程的优先级数值 setPriority（）：设置线程的优先级数值 Thread.sleep（

) : 将当前线程睡眠指定毫秒数 join () : 调用某线程的该方法，将当前线程与该线程"合并"，即等待该线程结束，再恢复当前线程的运行 yield () : 让出cpu，当前线程进入就绪队列等待调度 wait () : 当前线程进入对象的 wait pool notify () /notifyAll () : 唤醒对象的wait pool中的一个/所有等待线程

四 线程同步：实现生产者消费者问题来说明线程问题，举例如下所示：

```
/** * 生产者消费者问题 */ package
com.basic.thread. /** * * @version 2009-05-06 */ public class
ProducerConsumer { /** * @param args */ public static void
main(String[] args) { ProductBox pb = new ProductBox(). Producer
p = new Producer(pb). Consumer c = new Consumer(pb). Thread
pThread = new Thread(p). Thread cThread = new Thread(c).
pThread.setPriority(Thread.MAX_PRIORITY). pThread.start().
cThread.start(). } } /** * 产品对象 */ class Product { int id. public
Product(int id) { super(). this.id = id. } public String toString(){
return "Product : " id. } } /** * 产品盒对象 */ class ProductBox {
Product[] productbox = new Product[6]. int index = 0. public
ProductBox() { super(). } 100Test 下载频道开通，各类考试题目
直接下载。详细请访问 www.100test.com
```