创建线程

//1):定义一个类A继承于java.lang.Thread类.

class MusicThread extends Thread{

//2):在A类中覆盖Thread类中的run方法.

public void run() {

//3):在run方法中编写需要执行的操作

for(int i = 0; i < 50; i ++){

System.out.println("播放音乐"+i);

}

}

}

public class ExtendsThreadDemo {

public static void main(String[] args) {

for(int j = 0; j < 50; j ++){

System.out.println("运行游戏"+j);

if(j == 10){

//4):在main方法(线程)中,创建线程对象,并启动线程.

MusicThread music = new MusicThread();

music.start();

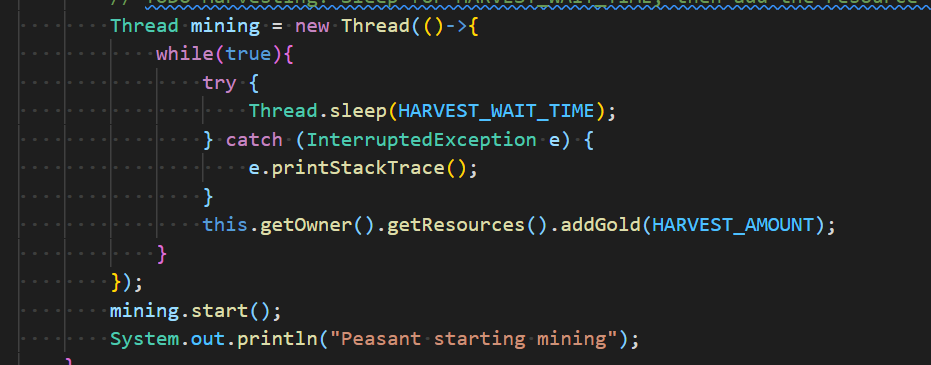
}

}

}

}

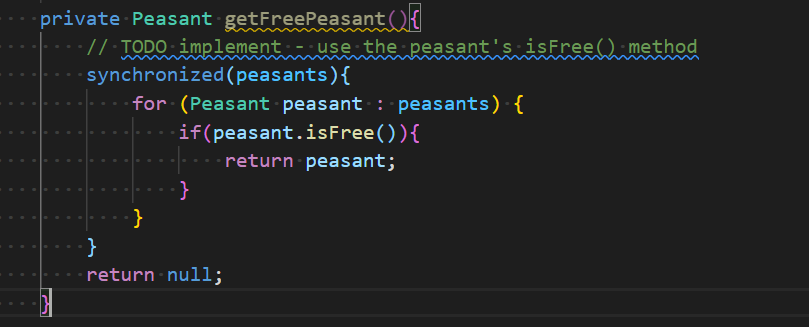
创建线程



线程池：



遍历线程安全的list时需要synchronized



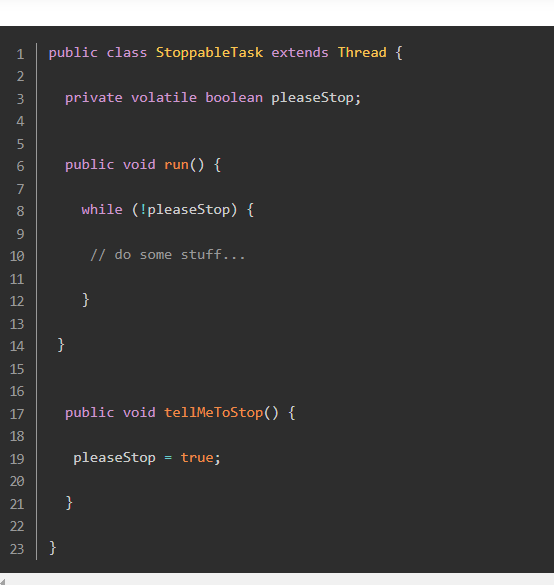
Java等待子线程

<https://cloud.tencent.com/developer/article/1802925>

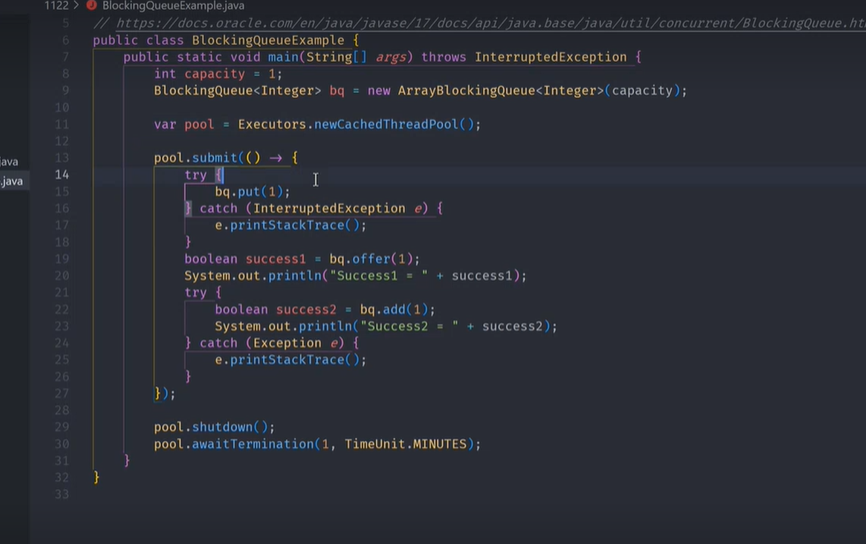
volatile 用法

<https://www.jianshu.com/p/ffb8e441b55b>

private volatile boolean pleaseStop;

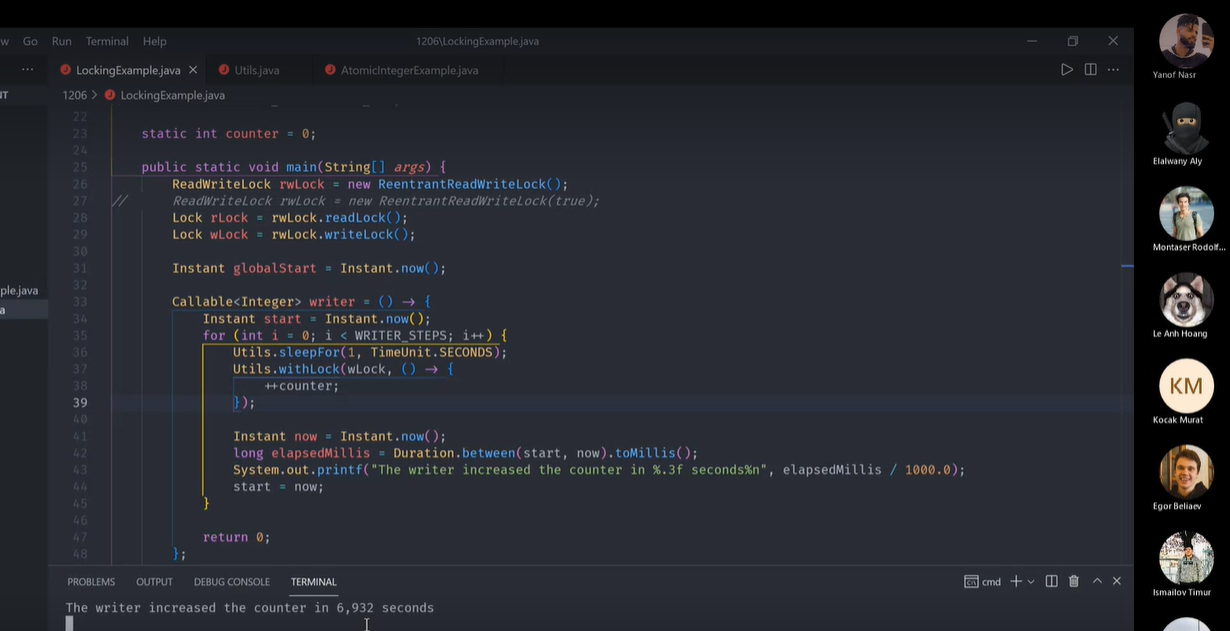


BlockingQueue

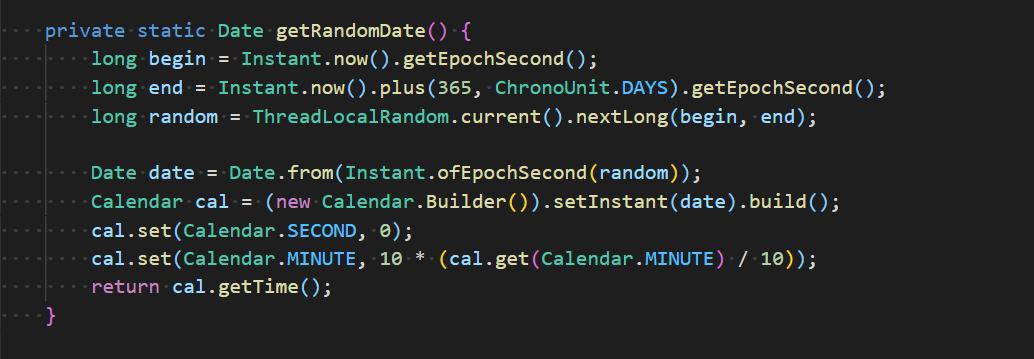


Add offer put方法区别

持续时间duration 和 读写锁的使用



RandomDate的使用



Callable的声明和使用

