一、编写一个继承Thread类的方式实现多线程的程序。该类MyThread有两个属性，一个字符串WhoAmI代表线程名，

一个整数delay代表该线程随机要休眠的时间。构造有参的构造器，线程执行时，显示线程名和要休眠时间。

另外，定义一个测试类TestThread，创建三个线程对象以展示执行情况。

class MyThread extends Thread{

private String whoAmI;

private int delay;

public MyThread(String s,int d){

whoAmI = s;

delay = d;

}

public void run(){

try{

sleep(delay);

}catch(InterruptedException ie){

}

System.out.println("Hello!I am"+whoAmI+",I slept"+delay+"milliseconds");

}

}

public class TestThread{

public static void main(String[] args){

MyThread t1 = new MyThread("Thread-1",(int)(Math.random()\*100));

MyThread t2 = new MyThread("Thread-2",(int)(Math.random()\*100));

MyThread t3 = new MyThread("Thread-3",(int)(Math.random()\*100));

t1.start();

t2.start();

t3.start();

}

}

二、利用多线程设计一个程序，同时输出 50 以内的奇数和偶数，以及当前运行的线程名。

public class Threadprint extends Thread {

int k = 1;

public void run() {

int i=k;

while(i<50) { System.out.println(Thread.currentThread().getName()+"-----"+i);

i+=2;

}

System.out.println(Thread.currentThread().getName()+" end!");

}

public static void main (String[] args) {

Threadprint t1=new Threadprint();

Threadprint t2=new Threadprint();

t1.k = 1;

t2.k = 2;

t1.start();

t2.start();

}

}

三、1、建立三个线程，A线程打印10次A，B线程打印10次B,C线程打印10次C，要求线程同时运行，交替打印10次ABC

**public class ABCThreadTest {**

**/\*\***

**\* @param args**

**\* @throws InterruptedException**

**\*/**

**public static void main(String[] args) throws InterruptedException {**

**Object a = new Object();**

**Object b = new Object();**

**Object c = new Object();**

**MyThreadPrinter pa = new MyThreadPrinter("A", c, a);**

**MyThreadPrinter pb = new MyThreadPrinter("B", a, b);**

**MyThreadPrinter pc = new MyThreadPrinter("C", b, c);**

**new Thread(pa).start();**

**Thread.sleep(120);**

**new Thread(pb).start();**

**Thread.sleep(120);**

**new Thread(pc).start();**

**Thread.sleep(120);**

**}**

**}**

**class MyThreadPrinter implements Runnable{**

**private String name;**

**private Object prev;**

**private Object self;**

**MyThreadPrinter (String name, Object prev, Object self) {**

**this.name = name;**

**this.prev = prev;**

**this.self = self;**

**}**

**@Override**

**public void run() {**

**int count = 10;**

**while (count > 0) {**

**synchronized (prev) {**

**synchronized (self) {**

**System.out.print(name);**

**count--;**

**try {**

**Thread.sleep(100);**

**} catch (InterruptedException e) {**

**e.printStackTrace();**

**}**

**self.notify();**

**}**

**try {**

**if(count>0) prev.wait();**

**} catch (InterruptedException e) {**

**e.printStackTrace();**

**}**

**}**

**}**

**}**

**}**

四、编程一个多线程程序，要求交叉输出100以内的奇偶数，具体如下：

1. 编程一个线程A输出100以内的奇数
2. 编程一个线程B输出100以内的偶数

要求线程A输出一个奇数数，接着线程B输出一个偶数，如此交叉输出100以内的所有奇偶数。

//要打印的资源

public class Num {

int i=1;

boolean flag = false; //两个线程，交替执行的一个标志

}

//打印奇数的线程

public class PrintQi implements Runnable{

Num num ;

public PrintQi(Num num)

{

this.num = num;

}

public void run()

{

while(num.i<= 100)

{

synchronized (num) {

if(num.flag)

{

try {

num.wait();

} catch (Exception e) {

}

}

else {

System.out.println("奇数"+num.i);

num.i++;

num.flag = true;

num.notify();

}

}

}

}

}

//打印偶数的线程

public class PrintOu implements Runnable{

Num num;

public PrintOu(Num num) {

this.num = num;

}

public void run()

{

while(num.i<=100)

{

synchronized (num)/\* 必须要用一把锁对象，这个对象是num\*/ {

if(!num.flag)

{

try

{

num.wait(); //操作wait()函数的必须和锁是同一个

} catch (Exception e)

{}

}

else {

System.out.println("oushu-----"+num.i);

num.i++;

num.flag = false;

num.notify();

}

}

}

}

}

//主函数

public class main {

public static void main(String[] args) {

Num num = new Num(); //声明一个资源

PrintQi pQi = new PrintQi(num);

PrintOu pOu = new PrintOu(num);

Thread aThread = new Thread(pQi);

Thread bThread = new Thread(pOu);

aThread.start();

bThread.start();

}

}