```java
public class ProducerConsumer{

    public static void main(String[] args) {

        MyBuffer b = new MyBuffer();

        Producer p1 = new Producer(b, 1);
        Consumer c1 = new Consumer(b, 1);

        p1.start();
        c1.start();
    }
}

class MyBuffer {
    private int contents;
    private boolean available = false;

    public synchronized int get(int who) {
        while (available == false) {
            try {
                wait();  //this.wait()
            } catch (InterruptedException e) { }
        }
        available = false;
        System.out.format("Consumer %d got: %d%n", who, contents);
        notifyAll();
        return contents;
    }
    public synchronized void put(int who, int value) {
        while (available == true) {
            try {
                wait();  //this.wait()
            } catch (InterruptedException e) { }
        }
        contents = value;
        available = true;
        System.out.format("Producer %d put: %d%n", who, contents);
        notifyAll();
    }
}

class Producer extends Thread {

    private MyBuffer buffer;
    private int number;

    public Producer(MyBuffer b, int number) {
        buffer = b;
        this.number = number;
    }

    public void run() {
        for (int i = 0; i < 10; i++) {
            buffer.put(number, i);
            try {
                sleep((int)(Math.random() * 100));
            } catch (InterruptedException e) { }
        }
    }
}

class Consumer extends Thread {

    private MyBuffer buffer;
    private int number;

    public Consumer(MyBuffer b, int number) {
        buffer = b;
        this.number = number;
    }

    public void run() {
        int value = 0;
        for (int i = 0; i < 10; i++) {
            value = buffer.get(number);
        }
    }
}
```