Aim:

Write a **Java** program that correctly implements **Producer Consumer** problem using the concept of **Inter Thread communication**.

Sample Input and Sample Output:

```
PUT:0
GET:0
PUT:1
GET:1
PUT:2
GET:2
PUT:3
GET:3
PUT:4
GET:4
PUT:5
GET:5
```

Note: Iterate the **while-loop** in **run()** method upto **5** times in **Producer** and **Consumer** Class.

Source Code:

ProdCons.java

```java
class Q
{
    int n;
    boolean statusFlag=false;
    synchronized void put(int n)
    {
        try
        {
            while(statusFlag)
            {
                wait();
            }
        }
        catch(InterruptedException e)
        {

        }
        this.n=n;
        System.out.println("PUT:"+n);
        statusFlag=true;
        notify();
    }
    synchronized int get()
    {
        try
        {
            while(!statusFlag)
            {
```

```
            wait();
        }
    }
    catch(InterruptedException e)
    {

    }
    statusFlag=false;
    System.out.println("GET:"+n);
    notify();
    return n;
    }
}
class Producer implements Runnable
{
    Q q;
    Producer(Q q)
    {
        this.q=q;
        new Thread(this,"Producer").start();
    }
    public void run()
    {
        int i=0;
        while(true)
        {
            q.put(i++);
            if(i==6)

                System.exit(0);

        }
    }
}
class Consumer implements Runnable
{
    Q q;
    Consumer(Q q)
    {
        this.q=q;
        new Thread(this,"Consumer").start();
    }
    public void run()
    {
        while(true)
        {
            q.get();
        }
    }
}
public class ProdCons
{
    public static void main(String args[])
    {
            Q q=new Q();
            Producer p=new Producer(q);
```

```
        Consumer c=new Consumer(q);
    }
}
```

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
PUT:0
GET:0
PUT:1
GET:1
PUT:2
GET:2
PUT:3
GET:3
PUT:4
GET:4
PUT:5
GET:5