Lab-8 and 10

Code:-

class DisplayThread extends Thread {

private String message;

private int interval;

private boolean running = true;

public DisplayThread(String message, int interval) {

this.message = message;

this.interval = interval;

}

public void run() {

while (running) {

System.out.println(message);

try {

Thread.sleep(interval);

} catch (InterruptedException e) {

e.printStackTrace();

}

}

}

public void stopThread() {

running = false;

}

}

public class ThreadExample {

public static void main(String[] args) {

DisplayThread bmsThread = new DisplayThread("BMS College of Engineering", 10000);

DisplayThread cseThread = new DisplayThread("CSE", 2000);

bmsThread.start();

cseThread.start();

System.out.println("Press Enter to stop the threads...");

try {

System.in.read();

} catch (Exception e) {

e.printStackTrace();

}

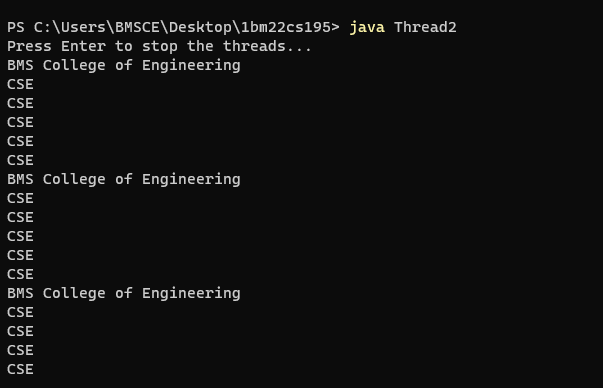
bmsThread.stopThread();

cseThread.stopThread();

}

}

Ouput for 8:-



Code for IPC(10):-

class Q {  
int n;  
boolean valueSet = false;  
synchronized int get() {  
while(!valueSet)  
try {  
wait();  
} catch(InterruptedException e) {  
System.out.println("InterruptedException caught");  
}  
System.out.println("Got: " + n);  
valueSet = false;  
notify();  
return n;  
}  
  
synchronized void put(int n) {  
while(valueSet)  
try {  
wait();  
} catch(InterruptedException e) {  
System.out.println("InterruptedException caught");  
}  
this.n = n;  
valueSet = true;  
System.out.println("Put: " + n);  
notify();  
}  
  
}  
class Producer implements Runnable {  
Q q;  
Producer(Q q) {  
this.q = q;  
new Thread(this, "Producer").start();  
}  
public void run() {  
int i = 0;  
while(i<15) {  
q.put(i++);  
}  
}  
}  
  
class Consumer implements Runnable {  
Q q;  
Consumer(Q q) {  
this.q = q;  
new Thread(this, "Consumer").start();  
}  
public void run() {  
int i=0;  
while(i<15) {  
int r=q.get();  
i++;  
}  
}  
}  
  
class PCFixed {  
public static void main(String args[]) {  
Q q = new Q();  
new Producer(q);  
new Consumer(q);  
System.out.println("Press Control-C to stop.");  
}  
}

output:-

