**Thread example 1:**

import java.io.\*;

import java.lang.\*;

public class ThreadExample

{

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

{

Thread t=Thread.currentThread();

System.out.println("Hey me .... : "+t);

t.setName("Demo Thread");

t.setPriority(2);

System.out.println("Waiting for end .... : "+t);

Thread.sleep(5000);

System.out.println(t.getPriority());

System.out.println(t.getName());

System.out.println(t.getState());

System.out.println(t.isAlive());

}

}

/\*try{

System.out.println("Hey me .... : "+t);

t.setName("Demo Thread");

t.setPriority(2);

System.out.println("Waiting for end .... : "+t.getName());

System.out.println("I am going to sleep ............ .... : "+t);

Thread.sleep(10000);

System.out.println("Byeeee.........");

}

catch(InterruptedException e)

{

System.out.println("Main Thread Interrupted");

}

System.out.println("Main Thread Exiting");

}

}\*/

import java.io.\*;

import java.lang.\*;

class NewThread implements Runnable

{

Thread t;

public NewThread()

{

t=new Thread(this,"Child Thread");

System.out.println("Thread Name : "+t);

t.start();

}

public void run()

{

try{

for(int i=5;i>0;i--)

{

System.out.println("Child Thread :"+i);

Thread.sleep(500);

}

}

catch(InterruptedException e)

{

System.out.println("Child Thread Interrupted");

}

System.out.println("Child is exiting");

}

}

public class ThreadExample1

{

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

{

NewThread nn=new NewThread();

System.out.println("Main Thread");

try{

for(int i=5;i>0;i--)

{

System.out.println("Main Thread :"+i);

Thread.sleep(1000);

}

}

catch(InterruptedException e)

{

System.out.println("Main Thread Interrupted");

}

System.out.println("Main is exiting");

}

}

**Thread example 2:**

import java.io.\*;

import java.lang.\*;

class NewThread extends Thread

{

public NewThread()

{

super("Child Thread");

System.out.println("Child Thread");

start();

}

public void run()

{

try{

for(int i=5;i>0;i--)

{

System.out.println("Child Thread :"+i);

Thread.sleep(5000);

}

}

catch(InterruptedException e)

{

System.out.println("Child Thread Interrupted");

}

System.out.println("Child is exiting");

}

}

public class ThreadExample2

{

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

{

NewThread n=new NewThread();

//n.start();

try{

for(int i=5;i>0;i--)

{

System.out.println("Main Thread :"+i);

Thread.sleep(10000);

}

}

catch(InterruptedException e)

{

System.out.println("Main Thread Interrupted");

}

System.out.println("Main is exiting");

}

}

**Thread example 3:**

import java.io.\*;

import java.lang.\*;

class NewThread implements Runnable

{

Thread t;

public NewThread(String n)

{

t=new Thread(this,n);

System.out.println("Thread Name : "+t.getName());

t.start();

}

public void run()

{

try{

for(int i=5;i>0;i--)

{

System.out.println("Child Thread : "+t.getName()+" :"+i);

Thread.sleep(5000);

}

}

catch(InterruptedException e)

{

System.out.println("Child Thread Interrupted");

}

System.out.println("Child "+t.getName() +" is exiting");

}

}

public class ThreadExample3

{

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

{

NewThread ob1=new NewThread("One");

NewThread ob2=new NewThread("Two");

NewThread ob3=new NewThread("Three");

NewThread ob4=new NewThread("Four");

/\*NewThread ob5=new NewThread("Five");

NewThread ob6=new NewThread("Six");

NewThread ob7=new NewThread("Seven");

NewThread ob8=new NewThread("Eight");

NewThread ob9=new NewThread("Nine");

NewThread ob10=new NewThread("Ten");\*/

try{

System.out.println("Waiting for End");

Thread.sleep(1000);

}

catch(InterruptedException e)

{

System.out.println("Main Thread Interrupted");

}

}

}

**Thread example 4:**

import java.io.\*;

import java.lang.\*;

class NewThread implements Runnable

{

Thread t;

public NewThread(String n)

{

t=new Thread(this,n);

System.out.println(" "+t.getState());

System.out.println("Thread Name : "+t.getName());

t.start();

}

public void run()

{

try{

for(int i=5;i>0;i--)

{

System.out.println("Child Thread "+t.getName()+" :"+i);

Thread.sleep(1000);

}

}

catch(InterruptedException e)

{

System.out.println("Child Thread Interrupted");

}

System.out.println("Child"+t.getName() +" is exiting");

}

}

public class ThreadExample4

{

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

{

NewThread ob1=new NewThread("One");

System.out.println("Now...."+ob1.t.getState());

NewThread ob2=new NewThread("Two");

NewThread ob3=new NewThread("Three");

System.out.println("Thread One status"+(ob1.t.isAlive()?" Alive":" No"));

System.out.println("Thread Two status"+(ob2.t.isAlive()?" Alive":" No"));

System.out.println("Thread Three status"+(ob3.t.isAlive()?" Alive":" No"));

try{

Thread.sleep(1000);

System.out.println("Thread One is going to sleep");

ob1.t.suspend();

System.out.println("Now...."+ob1.t.getState());

System.out.println("Thread One is going to resume the process");

ob1.t.resume();

System.out.println("Now...."+ob1.t.getState());

System.out.println("Waiting for End");

Thread.sleep(1000);

ob1.t.join();

System.out.println("One status "+ob1.t.getState());

ob2.t.join();

ob3.t.join();

}

catch(InterruptedException e)

{

System.out.println("Main Thread Interrupted");

}

System.out.println("Thread One status"+(ob1.t.isAlive()?" Alive":" No"));

System.out.println("Thread Two status"+(ob2.t.isAlive()?" Alive":" No"));

System.out.println("Thread Three status"+(ob3.t.isAlive()?" Alive":" No"));

}

}

**Thread example 5:**

import java.io.\*;

import java.lang.\*;

class ThreadExample5

{

public static void main(String args[])

{

Thread t=Thread.currentThread();

t.setName("Main Thread");

t.setPriority(Thread.MAX\_PRIORITY);

System.out.println("The Maximum Priority of Thread "+t.getName()+" is : "+t.getPriority());

t.setPriority(Thread.NORM\_PRIORITY);

System.out.println("The Normal Priority of Thread "+t.getName()+" is : "+t.getPriority());

t.setPriority(Thread.MIN\_PRIORITY);

System.out.println("The Minimum Priority of Thread "+t.getName()+" is : "+t.getPriority());

}

}

**Thread example 6:**

import java.io.\*;

import java.lang.\*;

class Queen

{

int n;

synchronized int get()

{

System.out.println("i got : "+this.n);

return this.n;

}

synchronized void put(int n)

{

this.n=n;

System.out.println("i set : "+this.n);

}

}

class Producer implements Runnable

{

Queen q;

Producer(Queen q)

{

this.q=q;

new Thread(this,"Producer").start();

}

public void run()

{

int i=0;

while(true)

q.put(i++);

}

}

class Consumer implements Runnable

{

Queen q;

Consumer(Queen q)

{

this.q=q;

new Thread(this,"Consumer").start();

}

public void run()

{

while(true)

q.get();

}

}

class ThreadExample6

{

public static void main(String args[])

{

Queen q=new Queen();

new Producer(q);

new Consumer(q);

System.out.println("Press Ctrl-c to stop");

}

}

**Thread example 7:**

import java.io.\*;

import java.lang.\*;

class Callme

{

void call(String m)

{

try

{

System.out.print("\n["+m);

System.out.print("]");

Thread.sleep(1000);

}

catch(InterruptedException e)

{

System.out.println("Child Thread Interrupted");

}

}

}

class NewThread implements Runnable

{

Thread t;

Callme ca;

String msg;

public NewThread(Callme c,String n,String m)

{

msg=m;

this.ca=c;

t=new Thread(this,n);

t.start();

}

public void run()

{

try{

Thread.sleep(1000);

synchronized(ca){

ca.call(msg);

}

}

catch(InterruptedException e)

{

System.out.println("Child Thread Interrupted");

}

}

}

public class ThreadExample7

{

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

{

Callme c1=new Callme();

NewThread ob1=new NewThread(c1,"One","Hello World");

NewThread ob2=new NewThread(c1,"Two","Say");

NewThread ob3=new NewThread(c1,"Three","Hi..");

NewThread ob4=new NewThread(c1,"Four","Good Bye");

try{

System.out.println("Waiting for End");

Thread.sleep(1000);

}

catch(InterruptedException e)

{

System.out.println("Main Thread Interrupted");

}

}

}

**Thread example 8:**

import java.io.\*;

import java.lang.\*;

class Queen

{

int n;

boolean status=false;

synchronized int get()

{

if(!status)

{

try

{

wait();

}

catch(InterruptedException e)

{

System.out.println("Child Thread Interrupted");

}

}

System.out.println("Iyyyyeeee i got : "+this.n);

status=false;

notify();

return this.n;

}

synchronized void put(int n)

{

if(status)

{

try

{

wait();

}

catch(InterruptedException e)

{

System.out.println("Child Thread Interrupted");

}

}

this.n=n;

status=true;

notify();

System.out.println("Hmmmmm i set : "+this.n);

}

}

class Producer implements Runnable

{

Queen q;

Producer(Queen q)

{

this.q=q;

new Thread(this,"Producer").start();

}

public void run()

{

int i=0;

while(true)

q.put(i++);

}

}

class Consumer implements Runnable

{

Queen q;

Consumer(Queen q)

{

this.q=q;

new Thread(this,"Consumer").start();

}

public void run()

{

while(true)

q.get();

}

}

class ThreadExample8

{

public static void main(String args[])

{

Queen q=new Queen();

new Producer(q);

new Consumer(q);

System.out.println("Press Ctrl-c to stop");

}

}