//EJEMPLO 1

**Ejemplo 1**

*public class Ejemplo1 {*

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

*HiloEjemplo1 hilo1 = new HiloEjemplo1("PEDRO");*

*HiloEjemplo1 hilo2 = new HiloEjemplo1("MARÍA");*

*HiloEjemplo1 hilo3 = new HiloEjemplo1("ALEX");*

*hilo1.start();*

*hilo2.start();*

*hilo3.start();*

*for(int i=0; i<25; i++){*

*System.out.println("Soy el main");*

*}*

*}*

*}*

*public class HiloEjemplo1 extends Thread {*

*private String nombreHilo;*

*public HiloEjemplo1(String nombreHilo) {*

*this.nombreHilo = nombreHilo;*

*}*

*@Override*

*public void run() {*

*for (int i = 0; i < 100; i++)*

*System.out.println("HOLA, SOY " + nombreHilo*

*+ "y voy por la iteración " + i);*

*}*

*}*

**Ejemplo 2**

*public class Ejemplo2 {*

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

*HiloEjemplo2 runneable1 = new HiloEjemplo2("PEDRO");*

*HiloEjemplo2 runneable2 = new HiloEjemplo2("MARÍA");*

*HiloEjemplo2 runneable3 = new HiloEjemplo2("ALEX");*

*Thread hilo1 = new Thread(runneable1);*

*Thread hilo2 = new Thread(runneable2);*

*Thread hilo3 = new Thread(runneable3);*

*hilo1.start();*

*hilo2.start();*

*hilo3.start();*

*for(int i=0; i<25; i++){*

*System.out.println("Soy el main");*

*}*

*}*

*}*

*public class HiloEjemplo2 implements Runnable {*

*private String nombreHilo;*

*public HiloEjemplo2(String nombreHilo) {*

*this.nombreHilo = nombreHilo;*

*}*

*@Override*

*public void run() {*

*for (int i = 0; i < 100; i++)*

*System.out.println("HOLA, SOY " + nombreHilo*

*+ "y voy por la iteración " + i);*

*}*

*}*

**Ejemplo 3**

*public class Ejemplo3 {*

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

*List <HiloEjemplo3> escritores = new ArrayList<HiloEjemplo3>();*

*escritores.add(new HiloEjemplo3(8, "Luis"));*

*escritores.add(new HiloEjemplo3(5, "Juan"));*

*escritores.add(new HiloEjemplo3(10, "Gloria"));*

*escritores.add(new HiloEjemplo3(9, "Maria"));*

*escritores.add(new HiloEjemplo3(7, "Alex"));*

*for(HiloEjemplo3 escritor : escritores)*

*escritor.start();*

*}*

*}*

*public class HiloEjemplo3 extends Thread {*

*private int tiempoEscritura;*

*public HiloEjemplo3(int tiempoEscritura, String nombre) {*

*this.tiempoEscritura = tiempoEscritura;*

*setName(nombre);*

*}*

*@Override*

*public void run() {*

*escribir();*

*}*

*public void escribir() {*

*System.out.println(getName() + " ha comenzado a escribir.");*

*try {*

*sleep(tiempoEscritura \* 1000);*

*} catch (InterruptedException e) {*

*// TODO Auto-generated catch block*

*e.printStackTrace();*

*}*

*System.out.println(getName() + " ha terminado de escribir.");*

*}*

*}*

**Ejemplo 4**

*public class Ejemplo4 {*

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

*HiloEjemplo3 escritor = new HiloEjemplo3(8, "Luis");*

*Set<Thread.State> estados = new HashSet<Thread.State>();*

*estados.add(escritor.getState());*

*escritor.start();*

*while (escritor.getState() != Thread.State.TERMINATED) {*

*estados.add(escritor.getState());*

*}*

*estados.add(escritor.getState());*

*System.out.println("LOS ESTADOS POR LOS QUE HA PASADO EL HILO SON: ");*

*for (Thread.State estado : estados)*

*System.out.println(estado);*

*}*

*}*

**Ejemplo 5**

*public class Ejemplo5 {*

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

*HiloEjemplo5 hilo = new HiloEjemplo5();*

*hilo.start();*

*try {*

*Thread.sleep(5000);*

*} catch (InterruptedException e) {*

*e.printStackTrace();*

*}*

*hilo.pararHilo();*

*try {*

*Thread.sleep(5000);*

*} catch (InterruptedException e) {*

*e.printStackTrace();*

*}*

*hilo.reanudarHilo();*

*try {*

*Thread.sleep(5000);*

*} catch (InterruptedException e) {*

*e.printStackTrace();*

*}*

*hilo.matarHilo();*

*}*

*}*

*public class HiloEjemplo5 extends Thread{*

*boolean hiloDetenido = false , hiloFinalizado = false;*

*@Override*

*public void run() {*

*while(!hiloFinalizado) {*

*if(hiloDetenido)*

*System.out.println("Estoy detenido");*

*else*

*System.out.println("Hilo en ejecución");*

*try {*

*sleep(1000);*

*} catch (InterruptedException e) {*

*e.printStackTrace();*

*}*

*}*

*}*

*public void matarHilo() {*

*hiloFinalizado = true;*

*}*

*public void pararHilo() {*

*hiloDetenido = true;*

*}*

*public void reanudarHilo() {*

*hiloDetenido = false;*

*}*

*}*

**Ejemplo 6**

*public class Ejemplo6 {*

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

*// TODO Auto-generated method stub*

*HiloEjemplo6 hilo = new HiloEjemplo6();*

*hilo.start();*

*try {*

*Thread.sleep(1);*

*} catch (InterruptedException e) {*

*// TODO Auto-generated catch block*

*e.printStackTrace();*

*}*

*hilo.interrupt();*

*}*

*}*

*public class HiloEjemplo6 extends Thread {*

*@Override*

*public void run() {*

*while (!isInterrupted()) {*

*System.out.println("Estoy en el hilo");*

*}*

*System.out.println("Finaliza la ejecución del hilo");*

*}*

*}*

**Ejemplo 7**

*public class Ejemplo7 {*

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

*Ejemplo7Contador contador = new Ejemplo7Contador(2000);*

*HiloEjemplo7Resta hiloResta = new HiloEjemplo7Resta(contador);*

*HiloEjemplo7Suma hiloSuma = new HiloEjemplo7Suma(contador);*

*System.out.println("El valor inicial del contador es " + contador.getValor());*

*hiloResta.start();*

*hiloSuma.start();*

*try {*

*hiloResta.join();*

*hiloSuma.join();*

*} catch (InterruptedException e) {*

*e.printStackTrace();*

*}*

*System.out.println("El valor final del contador es " + contador.getValor());*

*}*

*}*

*public class Ejemplo7Contador {*

*int valor;*

*public Ejemplo7Contador(int valor) {*

*this.valor = valor;*

*}*

*public synchronized void incrementarValor() {*

*valor++;*

*}*

*public synchronized void decrementarValor() {*

*valor--;*

*}*

*public int getValor() {*

*return valor;*

*}*

*}*

*public class HiloEjemplo7Resta extends Thread{*

*Ejemplo7Contador contador;*

*public HiloEjemplo7Resta(Ejemplo7Contador contador) {*

*this.contador = contador;*

*}*

*@Override*

*public void run() {*

*for(int i = 0; i < 1000; i++)*

*contador.decrementarValor();*

*}*

*}*

*public class HiloEjemplo7Suma extends Thread{*

*Ejemplo7Contador contador;*

*public HiloEjemplo7Suma(Ejemplo7Contador contador) {*

*this.contador = contador;*

*}*

*@Override*

*public void run() {*

*for(int i = 0; i < 1000; i++)*

*contador.incrementarValor();*

*}*

*}*

**Ejemplo 8**

*public class Ejemplo8 {*

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

*// TODO Auto-generated method stub*

*Object testigo = new Object();*

*HiloEjemplo8 hilo1 = new HiloEjemplo8("PING\t", testigo, Ejemplo8TipoImpresion.PING);*

*HiloEjemplo8 hilo2 = new HiloEjemplo8("PONG\n", testigo, Ejemplo8TipoImpresion.PONG);*

*hilo2.start();*

*hilo1.start();*

*}*

*}*

*public class HiloEjemplo8 extends Thread {*

*String textoAImprimir;*

*Object testigo;*

*Ejemplo8TipoImpresion tipoImpresion;*

*public HiloEjemplo8(String textoAImprimir, Object testigo, Ejemplo8TipoImpresion tipoImpresion) {*

*this.textoAImprimir = textoAImprimir;*

*this.testigo = testigo;*

*this.tipoImpresion = tipoImpresion;*

*}*

*@Override*

*public void run() {*

*for (int i = 0; i < 10; i++) {*

*synchronized (testigo) {*

*if (tipoImpresion == Ejemplo8TipoImpresion.PING) {*

*System.out.print(textoAImprimir);*

*testigo.notify();*

*try {*

*testigo.wait();*

*} catch (InterruptedException e) {*

*// TODO Auto-generated catch block*

*e.printStackTrace();*

*}*

*}else {*

*try {*

*testigo.wait();*

*} catch (InterruptedException e) {*

*// TODO Auto-generated catch block*

*e.printStackTrace();*

*}*

*System.out.print(textoAImprimir);*

*testigo.notify();*

*}*

*}*

*}*

*}*

*}*

**Ejemplo 9**

*public class Ejemplo9 {*

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

*Ejemplo9Cola\_V2 cola = new Ejemplo9Cola\_V2();*

*Ejemplo9Productor productor = new Ejemplo9Productor(cola);*

*Ejemplo9Consumidor consumidor = new Ejemplo9Consumidor(cola);*

*productor.start();*

*consumidor.start();*

*}*

*}*

*public class Ejemplo9Cola {*

*private int numero;*

*private boolean disponible = false;*

*public int get(){*

*if(disponible){*

*disponible = false;*

*return numero;*

*}*

*return -1;//No hay número disponible, cola vacía.*

*}*

*public void put(int valor){*

*numero = valor;*

*disponible = true;*

*}*

*}*

*public class Ejemplo9Cola\_V2 {*

*private int numero;*

*private boolean disponible = false;*

*public synchronized int get() {*

*while (!disponible) {*

*try {*

*wait();*

*} catch (InterruptedException ex) {*

*System.out.println("Interrupción.");*

*}*

*}*

*disponible = false;*

*notify();*

*System.out.println(" Consumidor :" + numero + " consume: " + numero);*

*return numero;*

*}*

*public synchronized void put(int valor) {*

*while (disponible) {*

*try {*

*wait();*

*} catch (InterruptedException ex) {*

*System.out.println("Interrupción.");*

*}*

*}*

*numero = valor;*

*disponible = true;*

*System.out.println("EL productor : produce: " + valor);*

*notify();*

*}*

*}*

*ublic class Ejemplo9Consumidor extends Thread{*

*private final Ejemplo9Cola\_V2 cola;*

*public Ejemplo9Consumidor(Ejemplo9Cola\_V2 cola) {*

*this.cola = cola;*

*}*

*@Override*

*public void run() {*

*int valor;*

*for (int i = 0; i < 5; i++) {*

*valor = cola.get();*

*//No hemos puesto pausa, por tanto, el consumidor será más rápido que el productor.*

*}*

*}*

*}*

*public class Ejemplo9Productor extends Thread {*

*private final Ejemplo9Cola\_V2 cola;*

*public Ejemplo9Productor(Ejemplo9Cola\_V2 cola) {*

*this.cola = cola;*

*}*

*@Override*

*public void run() {*

*for (int i = 0; i < 5; i++) {*

*cola.put(i);*

*try {*

*sleep(100);*

*} catch (InterruptedException ex) {*

*System.out.println("Interrupción.");*

*}*

*}*

*}*

*}*

**Ejemplo 10**

*public class Ejemplo10 {*

*final static int numeroDeProbadores = 5;*

*final static int numeroDeClientes = 100;*

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

*Semaphore semaforo = new Semaphore(numeroDeProbadores);*

*List<Ejemplo10Hilos> clientes = new ArrayList<Ejemplo10Hilos>();*

*for (int i = 0; i < numeroDeClientes; i++)*

*clientes.add(new Ejemplo10Hilos(i + 1));*

*Ejemplo10Hilos.setSemaforo(semaforo);*

*for (Ejemplo10Hilos cliente : clientes)*

*cliente.start();*

*for (Ejemplo10Hilos cliente : clientes)*

*try {*

*cliente.join();*

*} catch (InterruptedException e) {*

*// TODO Auto-generated catch block*

*e.printStackTrace();*

*}*

*System.out.println("TODOS LOS CLIENTES SE HAN PROBADO LA ROPA");*

*}*

*}*

*public class Ejemplo10Hilos extends Thread {*

*private int tiempoNecesitado;*

*private static Semaphore semaforo;*

*int id;*

*public Ejemplo10Hilos(int id) {*

*this.tiempoNecesitado = (int) (Math.random() \* 2000 + 1000);*

*this.id = id;*

*}*

*@Override*

*public void run() {*

*try {*

*semaforo.acquire();*

*usarProbador();*

*} catch (InterruptedException e) {*

*// TODO Auto-generated catch block*

*e.printStackTrace();*

*}*

*semaforo.release();*

*}*

*private void usarProbador() {*

*System.out.println("Soy el cliente " + id + " y voy a usar un probador.");*

*try {*

*sleep(tiempoNecesitado);*

*} catch (InterruptedException e) {*

*// TODO Auto-generated catch block*

*e.printStackTrace();*

*}*

*System.out.println("Soy el cliente " + id + " y salgo del probador.");*

*}*

*public static void setSemaforo(Semaphore semaforo) {*

*Ejemplo10Hilos.semaforo = semaforo;*

*}*

*}*