**ACTIVIDAD 1**

**Enunciado**

Realiza el siguiente ejercicio utilizando una única clase Java (Hilo) que sustituya a las clases Primero y Segundo.

El nombre de la clase principal será Actividad1 y el número de veces que se sacará por la consola las cadenas de caracteres “Primero” y “Segundo” será 20.

**//Hilos1.java**

**public** **class** Hilos1 {

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

Primero p=**new** Primero();

Segundo s=**new** Segundo();

p.start();

s.start();

System.***out***.print("Fin programa");

}

}

**//Primero.java**

**public class** Primero **extends** Thread {

**public** **void** run(){

**for** (**int** i=1;i<100;i++)

System.***out***.println("Primero " + i);

}

}

**//Segundo.java**

**public** **class** Segundo **extends** Thread {

**public** **void** run(){

**for** (**int** i=1;i<100;i++)

System.***out***.println("Segundo " + i);

}

}

**Solución**

**//Hilo.java**

public class Hilo extends Thread { //Extiende de Thread

public void run()

{

for (int i = 1 ; i <= 20; i++) //bucle de 20 repeticiones para sacar el nombre del hilo + su numero por pantalla

System.out.println(getName()+"\t" + i);

}

}

**//Actividad1.java**

public class Actividad1 {

public static void main(String[] args) {

Hilo hilo1= new Hilo(); //Creamos el primer hilo

hilo1.setName("Primero");//Le asignamos el nombre "Primero"

Hilo hilo2= new Hilo(); //Creamos el segundo hilo

hilo2.setName("Segundo"); //Le asignamos el nombre 'Segundo'

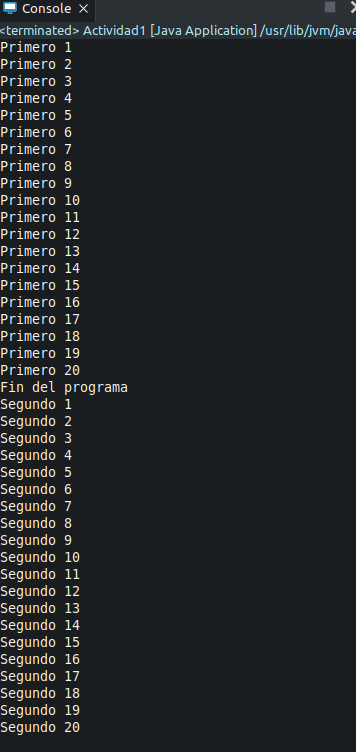
hilo1.start();

hilo2.start();//Los ejecutamos

System.out.println("Fin del programa"); //Fin del programa

}

}

****

**Resolución de problemas**

**ACTIVIDAD 2**

**Enunciado**

Realiza un programa que cree n hilos, donde n se pasará como argumento.

Por cada hilo creado, se sacará por consola 20 veces un mensaje del tipo: “Hilo n”, donde n será el número de hilo que se está ejecutando.

Después de haber creado los n hilos, sacar por consola un mensaje que diga “Final Programa”.

El nombre de la clase principal será Actividad2.

**Solución**

**//Hilo.java**

public class Hilo extends Thread { //Extiende de Thread

public void run()

{

for (int i = 1 ; i <= 20; i++) //bucle de 20 repeticiones para sacar el nombre del hilo + su numero por pantalla

System.out.println(getName()+"\t" + i);

}

}

**//Actividad2.java**

public class Actividad2 {

public static void main(String[] args) {

for (int i = 1; i <= Integer.parseInt(args[0]); i++) {

Hilo hilo = new Hilo(); //Creamos un hilo

hilo.setName("Hilo "+ i); //Le asignamos el nombre 'Hilo' + su numero

hilo.start(); //Ejecutamos

try {

hilo.join();//Esperamos a que termine

} catch (InterruptedException e) {

// TODO Auto-generated catch block

e.printStackTrace();

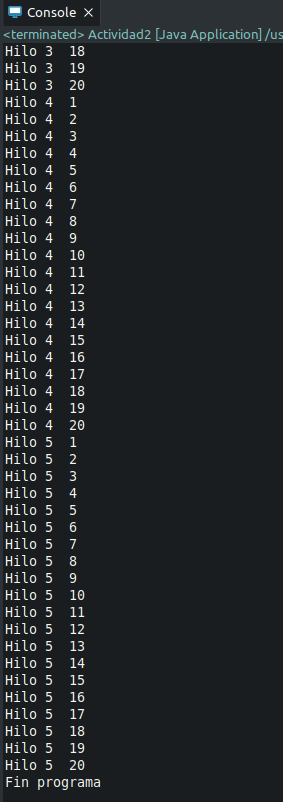
}

}

System.out.println("Fin programa"); //Fin del programa

}

}

****

**Resolución de problemas**

**ACTIVIDAD 3**

**Enunciado**

Realiza un programa que use los métodos setName() y setPriority() de la clase Thread para cambiar el nombre y prioridad iniciales del hilo. Además, será necesario utilizar los métodos getName() y getPriority() de la clase Thread para visualizar en consola, tanto los valores iniciales del nombre y prioridad del hilo como los valores posteriormente modificados.

El nombre de la clase principal será Actividad3.

El resultado esperado será el siguiente:

El nombre del hilo es Thread-0 y tiene la prioridad 5

Ahora el nombre del hilo es SUPER-HILO-DM2 y tiene la prioridad 6

Final programa

**Solución**

**//Hilo.java**

public class Hilo extends Thread{

public void run()

{

try {

sleep(10000); //Sleep de 10 segundos

} catch (InterruptedException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

**//Actividad3.java**

public class Actividad3 {

public static void main(String[] args) {

Hilo hilo = new Hilo(); //Creamos el hilo

hilo.setName("Thread-0"); //Le asignamos un nombre

hilo.setPriority(5); //le ponemos una prioridad

hilo.start(); //Lo ejecutamos

System.out.println("El nombre del hilo es " + hilo.getName() + " y tiene la prioridad " + hilo.getPriority());

hilo.setName("SUPER-HILO-DM2"); //Le cambiamos el nombre

hilo.setPriority(6); //Le cambiamos la prioridad

System.out.println("Ahora el nombre del hilo es " + hilo.getName() + " y tiene la prioridad " + hilo.getPriority());

System.out.print("Final Programa\n");

}

}

**Resolución de problemas**

**REQUISITOS**

1. Entregar las soluciones de las actividades en un único archivo comprimido donde tendremos:
   1. Un archivo de extensión .odt o .docx que contendrá
      1. Una copia del código del programa
      2. Una captura de pantalla del resultado de la ejecución del programa
   2. Los archivos \*.java empleados
2. El único archivo con la soluciones contendrá el nombre original de la actividad, seguido de un guión bajo y del primer apellido del alumno.

**CRITERIOS DE CORRECCIÓN**

1. Cumplimiento de los requisitos (15%).
2. Hacer lo que se indica en el enunciado (55%).
3. Claridad del código Java (15%).
4. Documentación del código Java (15%).
5. Explicación detallada de cómo se han resuelto problemas que hayan podido surgir en la realización de la actividad (hasta un 15% adicional sobre la nota obtenida). Dicha explicación se incluirá al principio de los archivos de código fuente Java mediante líneas de comentarios. La primera línea de comentarios sólo incluirá lo siguiente: // Resolución de problemas.
6. Variables no usadas (-15%).
7. Librerías no usadas (-15%).