**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**

**//Actividad1.java**

public class Actividad1 {

public static void main(String[] args) {

Hilo p=new Hilo(); //Creacion del primer hilo

p.setName("Primero"); //Le ponemos de nombre "Primero"

Hilo s=new Hilo(); //Creacion del segundo hilo

s.setName("Segundo"); //Le ponemos de nombre "Segundo"

p.start(); //Ejecutamos el primero

s.start(); //Ejecutamos el segundo

System.out.print("Fin programa\n"); //imprimimos por pantalla "Fin del programa"

}

}

**//Hilo.java**

public class Hilo extends Thread{

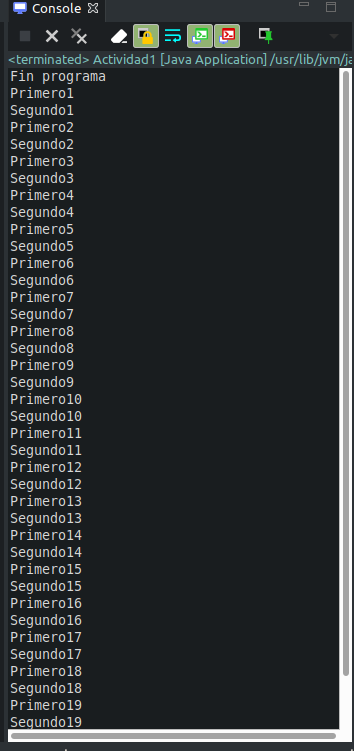
public void run(){

for (int i=1;i<=20;i++)

System.out.println(getName()+ i); //Imprimimos el nombre del hilo + el numero de repeticion

}

}



**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**

**//Actividad2.java**

public class Actividad2 {

public static void main(String[] args) {

for (int cont=1; cont<= Integer.parseInt(args[0]); cont++) { //bucle con n veces pasadas por argumento

Hilo hilo = new Hilo(); //Creacion del hilo

hilo.setName("Hilo " + cont + "\t"); //Se le cambia el nombre por le numero de hilo correspondiente

hilo.start(); //ejecutamos el hilo

try {

hilo.join(); //Espera a que todos los demas hilos terminen de ejecutarse

} catch (InterruptedException e) {

e.printStackTrace();

}

}

System.out.println("Fin programa\n");

}

}

**//Hilo.java**

public class Hilo extends Thread{

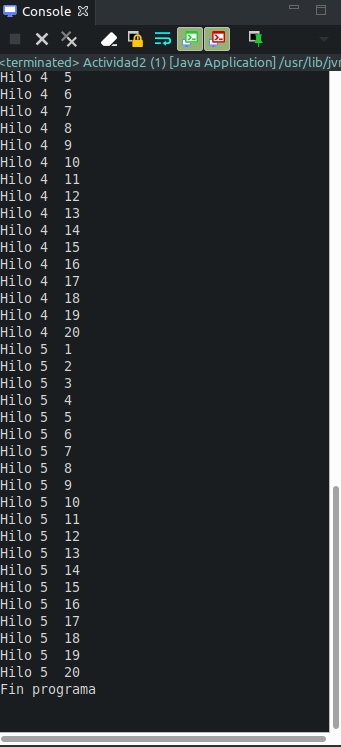
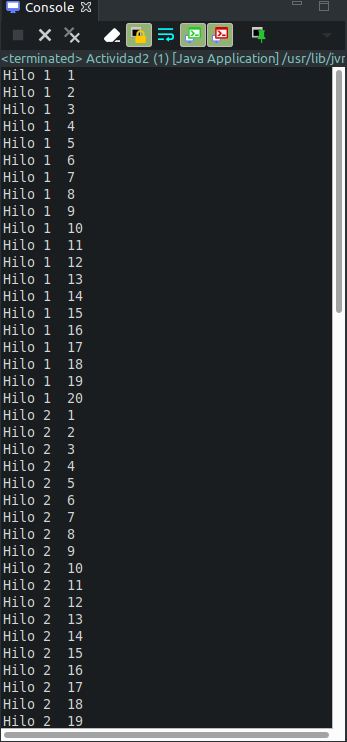
public void run(){

for (int i=1;i<=20;i++)

System.out.println(getName()+ i); //Imprimimos el nombre del hilo + el numero de repeticion

}

}

****

**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**

**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%).