

Programación de Servicios y Procesos

UT1: Práctica 3

Alexis López Briongos Dam2t

19/10/2023

Índice

1. Genera un programa que guarde en un fichero de nombre SVCHOST.TXT los servicios que se están ejecutando bajo el proceso svchost.exe. 2

1. Genera un programa que guarde en un fichero de nombre SVCHOST.TXT los servicios que se están ejecutando bajo el proceso svchost.exe.

Pista: String comando = "tasklist /svc /fi \"imagenname eq svchost.exe\""

```
public class Practica3 {  
  
    static Process p;  
    static BufferedReader reader;  
    static PrintWriter writer;  
    static String comando;
```

- Creamos las variables necesarias con las que vamos a operar en nuestro programa.

```
public static void main(String[] args) {  
  
    //Guardamos el comando que vamos a ejecutar para guardar el resultado en un fichero  
    comando = "tasklist /svc /fi \"imagenname eq svchost.exe\"";
```

- Inicializamos la variable comando con el comando que vamos a ejecutar para obtener las tareas asociadas con el servicio svchost.exe

```
//Try catch para controlar las excepciones  
try {  
  
    //Inicializamos el printwriter creando dentro de el un fileoutput para guardar  
    //el resultado de la ejecucion del comando  
    writer = new PrintWriter(new FileOutputStream("SVCHOST.TXT"));  
  
    //Ejecutamos el comando y obtenemos el proceso  
    p = Runtime.getRuntime().exec(comando);  
  
    //Inicializamos un reader para leer todas las lineas del resultado  
    reader = new BufferedReader(new InputStreamReader(p.getInputStream()));  
    String line;
```

- Dentro del try vamos a inicializar tanto las variables de lectura, escritura y obtención del proceso que le pasamos (variable comando)
- Creamos una variable de tipo String denominada "line" para almacenar las líneas leídas del resultado del comando.

```
String line;  
  
while ((line = reader.readLine()) != null) {  
    // Escribe el resultado linea por linea hasta que sea nula en el fichero creado  
    writer.print(line);  
}  
  
//Cerramos los readers y los writers  
reader.close();  
writer.close();
```

- Creamos un bucle almacenando las líneas leídas y comprobando si son nulas o no hasta que se encuentre con una línea nula.
- Con el `writer.print(line)` vamos añadiendo línea por línea en el fichero creado el resultado del proceso.
- Cerramos el `writer` y el `reader`.

```

    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
        System.out.println("Se ha originado un error");
    }
}

```

- Por último controlamos con un `catch` las excepciones devolviendo un error por consola.

```

public class Practica3 {

    static Process p;
    static BufferedReader reader;
    static PrintWriter writer;
    static String comando;

    public static void main(String[] args) {

        //Guardamos el comando que vamos a ejecutar para guardar el resultado en un fichero
        comando = "tasklist /svc /fi \"imagename eq svchost.exe\"";

        //Try catch para controlar las excepciones
        try {

            //Iniciamos el printwriter creando dentro de el un fileoutput para guardar
            //el resultado de la ejecución del comando
            writer = new PrintWriter(new FileOutputStream("SVCHOST.TXT"));

            //Ejecutamos el comando y obtenemos el proceso
            p = Runtime.getRuntime().exec(comando);

            //Iniciamos un reader para leer todas las líneas del resultado
            reader = new BufferedReader(new InputStreamReader(p.getInputStream()));
            String line;

            while ((line = reader.readLine()) != null) {
                // Escribimos el resultado línea por línea hasta que sea nula en el fichero creado
                writer.print(line);
            }

            //Cerramos los readers y los writers
            reader.close();
            writer.close();

        } catch (Exception e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
            System.out.println("Se ha originado un error");
        }
    }
}

```

- Imagen de la clase completa.