**ACTIVIDAD 11**

**Enunciado**

Realiza un programa Java (**Actividad11**) que ejecute el programa que se muestra a continuación (**Ejemplo2**). El programa Actividad11 sólo requiere adaptar el valor que se le asigna a la variable *comando*. En ambos programas se usan las clases Java Runtime y Processpara la gestión de procesos. Incluye los dos programas Java en el paquete por defecto que ofrece el IDE Eclipse.

Compila tanto el programa del enunciado (Ejemplo2) como el nuevo programa (Actividad11) desde la línea de comandos, aportando evidencias de cómo lo has hecho.

Después de que la compilación esté libre de errores, ejecuta el programa Java de nombre Actividad11 desde la línea de comandos, aportando evidencias de tanto su ejecución como del resultado de misma.

import java.io.\*;

public class Ejemplo2 {

public static void main(String[] args){

Runtime r=Runtime.*getRuntime*();

//String comando="ls -al"; //Descomentar la línea para Linux

//String comando="CMD C/ DIR"; //Descomentar la línea para Windows

Process p=null;

try {

p = r.exec (comando);

InputStream is = p.getInputStream();

BufferedReader br = new BufferedReader (new InputStreamReader(is));

String linea;

while((linea = br.readLine()) != null) // lee una linea

System.*out*.println(linea);

br.close();

}

catch (Exception e) {

e.printStackTrace();

}

///// Comprobación: Valor= 0 bien, Valor = - 1 mal

int exitVal;

try {

exitVal=p.waitFor();

System.*out*.println ("Valor de Salida "+exitVal);

} catch (InterruptedException e){

e.printStackTrace();

}

}

}

**Solución:**

package actividad11;

import java.io.\*;

public class Actividad11 {

public static void main(String[] args) {

Runtime r = Runtime.getRuntime();

String comando = "java Ejemplo2.java"; // Cambiamos el valor asignado a la variable comando para ejecutar Ejemplo2

Process p = null;

try {

p = r.exec(comando);

InputStream is = p.getInputStream();

BufferedReader br = new BufferedReader(new InputStreamReader(is));

String linea;

while ((linea = br.readLine()) != null) // lee una línea

System.out.println(linea);

br.close();

} catch (Exception e) {

e.printStackTrace();

}

// Comprobación: Valor= 0 bien, Valor = -1 mal

int exitVal;

try {

exitVal = p.waitFor();

System.out.println("Valor de Salida " + exitVal);

} catch (InterruptedException e) {

e.printStackTrace();

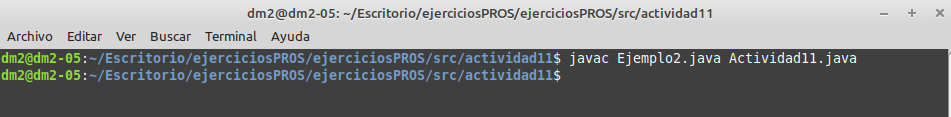
}

}

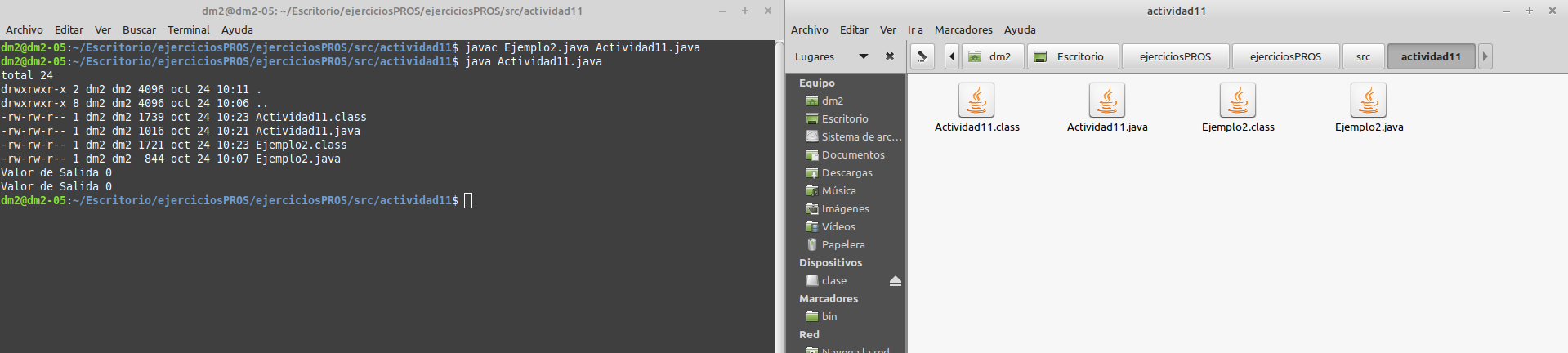
}

**Evidencias de la solución:**

* Compilación desde la línea de comandos de los programas fuente Ejemplo2.java y Actividad11.java:



* Ejecución del programa Actividad11.class desde la línea de comandos:

****

**Resolución de problemas:**

Al ejecutar Actividad11 me daba un valor de salida que indicaba un error, entonces tras unas pruebas ejecutando Ejemplo2 y mirando el codigo, me di cuenta que tenia que cambiar el valor de la variable “comando” del archivo Actividad11 y ponerle .java al final y asi podremos ejecutar desde la carpeta donde estan los .java y los .class usando “java Actividad11.java” en la consola ya que si ponemos solo “java Actividad11” da un error

**CRITERIOS DE CORRECCIÓN**

1. Nivel de cumplimiento de los Requisitos (15%).
2. Correcto funcionamiento del programa desarrollado en Java (25%).
3. Validez de las evidencias aportadas (60%).
4. 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).