

PASO DE PARÁMETROS POR LA LÍNEA DE COMANDOS

Java Básico

La frase del día

2



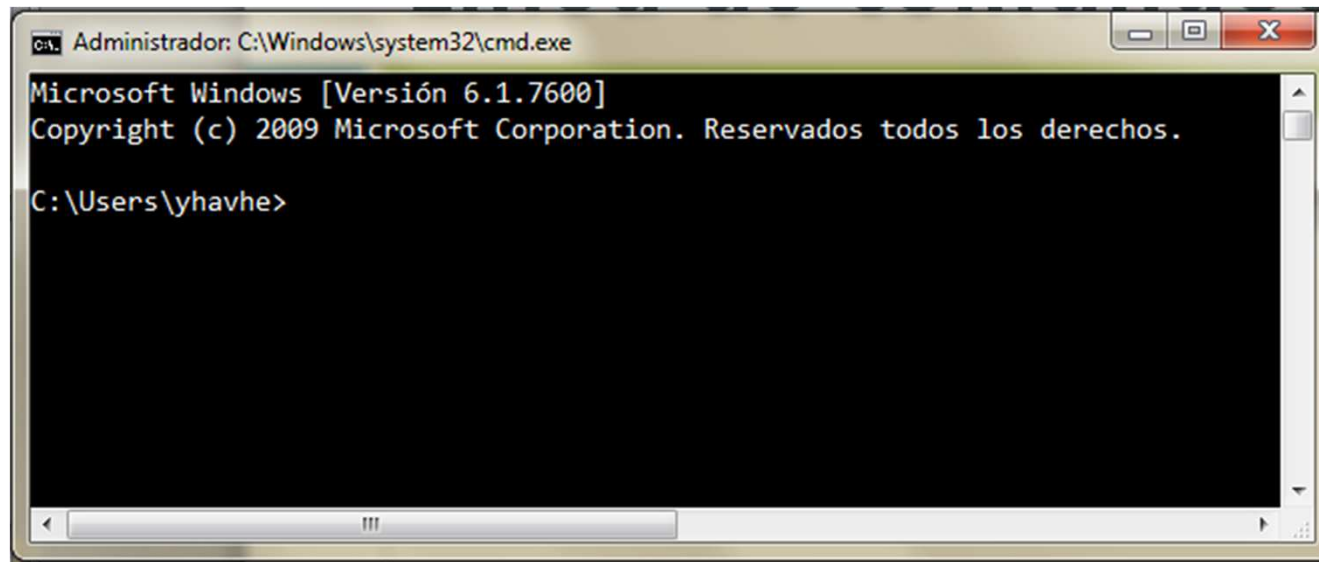
“When a programming language is created that allows programmers to program in simple English, it will be discovered that programmers cannot speak English.”

- *Anonymous*

Línea de comandos

3

- Tipo de interfaz para manipular un programa o sistema operativo con instrucciones escritas.

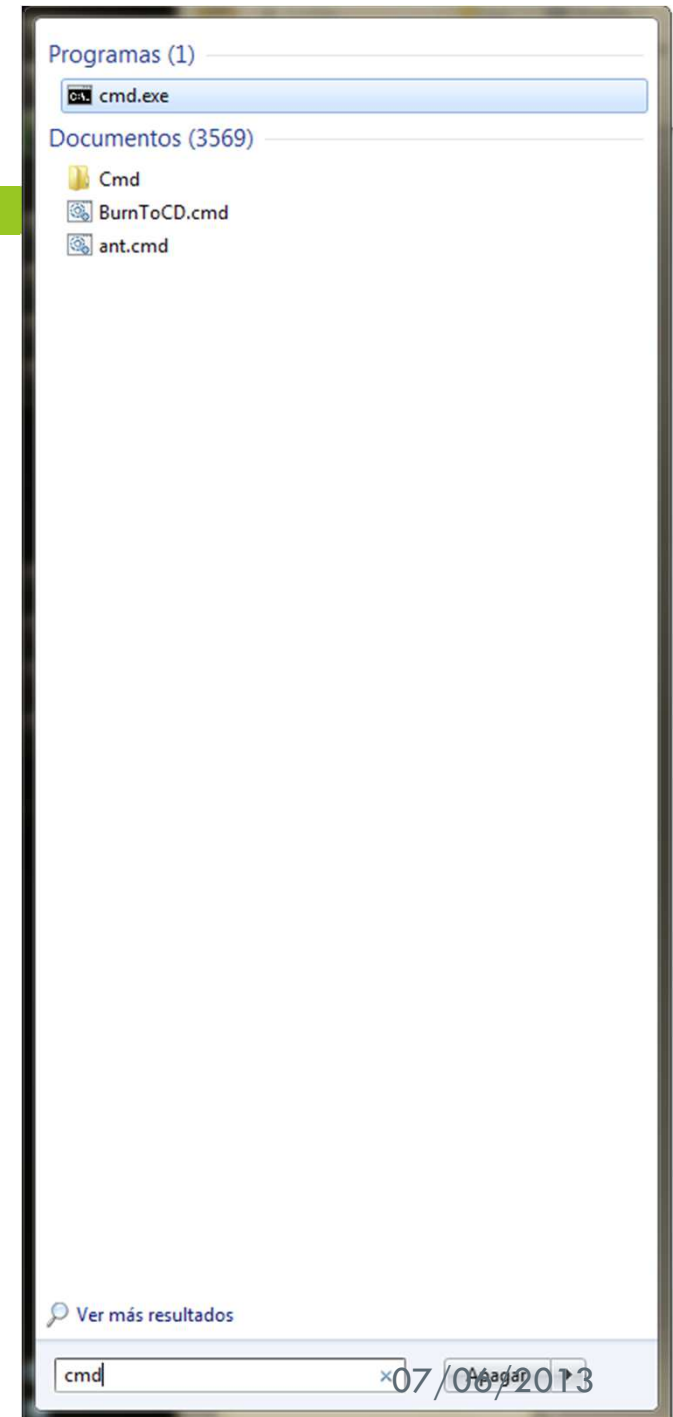


En windows

4

- Windows 7
 - Click en botón inicio
 - Buscar cmd
 - ENTER

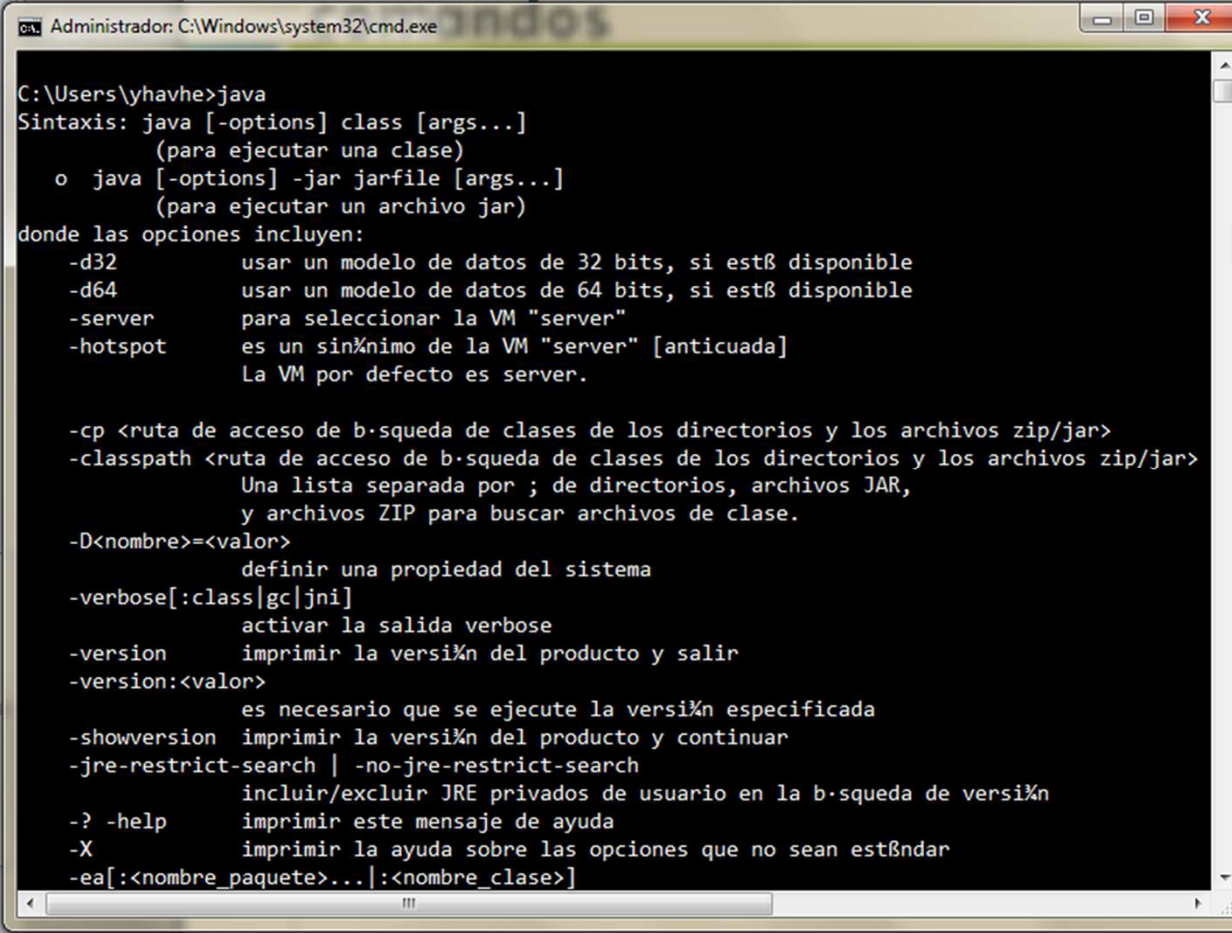
- Windows 8
 - Presionar botón window + R
 - Escribir cmd
 - ENTER



Ejecutando programas en la línea de comandos

5

□ Escribimos: `java`



```
Administrador: C:\Windows\system32\cmd.exe

C:\Users\yhavhe>java
 Sintaxis: java [-options] class [args...]
             (para ejecutar una clase)
   o  java [-options] -jar jarfile [args...]
       (para ejecutar un archivo jar)
donde las opciones incluyen:
  -d32      usar un modelo de datos de 32 bits, si est  disponible
  -d64      usar un modelo de datos de 64 bits, si est  disponible
  -server   para seleccionar la VM "server"
  -hotspot   es un sin nimo de la VM "server" [anticuada]
             La VM por defecto es server.

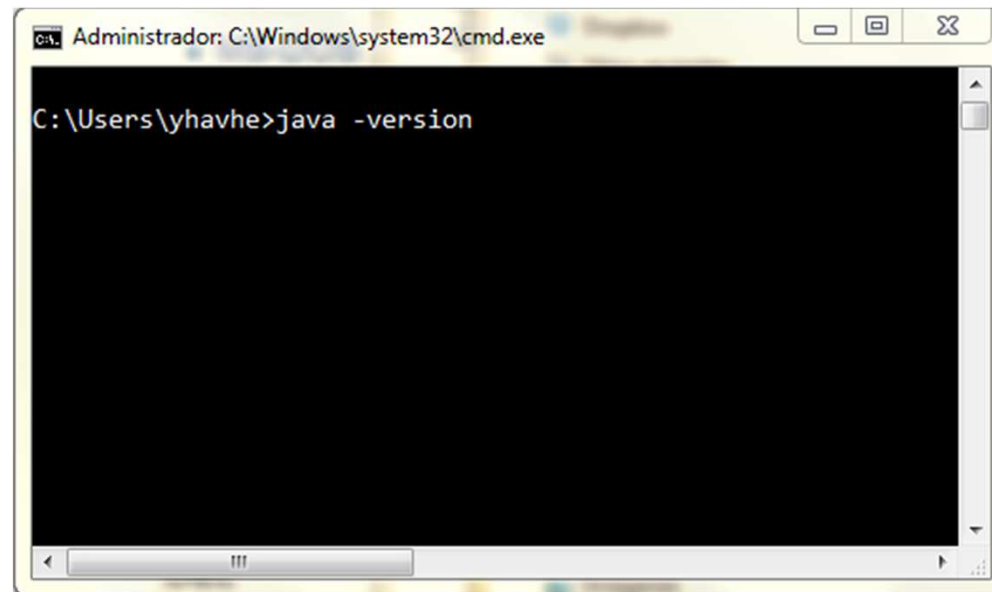
  -cp <ruta de acceso de b squeda de clases de los directorios y los archivos zip/jar>
  -classpath <ruta de acceso de b squeda de clases de los directorios y los archivos zip/jar>
             Una lista separada por ; de directorios, archivos JAR,
             y archivos ZIP para buscar archivos de clase.
  -D<nombre>=<valor>
             definir una propiedad del sistema
  -verbose[:class|gc|jni]
             activar la salida verbose
  -version   imprimir la versi n del producto y salir
  -version:<valor>
             es necesario que se ejecute la versi n especificada
  -showversion
             imprimir la versi n del producto y continuar
  -jre-restrict-search | -no-jre-restrict-search
             incluir/excluir JRE privados de usuario en la b squeda de versi n
  -? -help   imprimir este mensaje de ayuda
  -X         imprimir la ayuda sobre las opciones que no sean est ndar
  -ea[:<nombre_paquete>...|:<nombre_clase>]
```

Ejecutando programas en la línea de comandos

6

□ Escribimos:

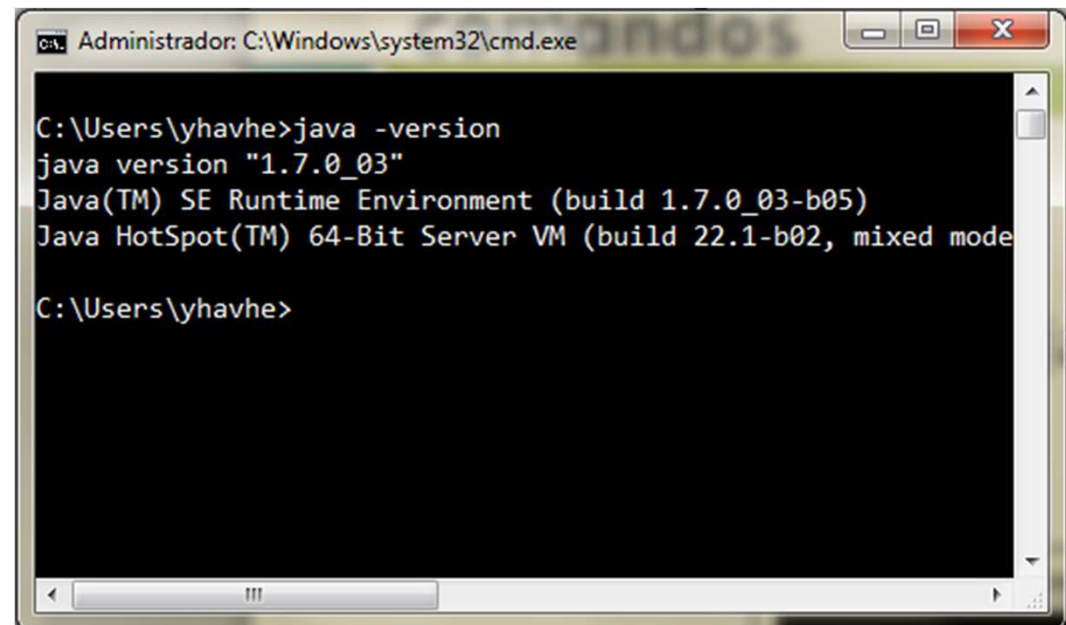
```
java -version
```



Ejecutando programas en la línea de comandos

7

- El programa java fue ejecutado con el parámetro
 - -version
- Algunos programas pueden recibir parámetros.



```
Administrador: C:\Windows\system32\cmd.exe

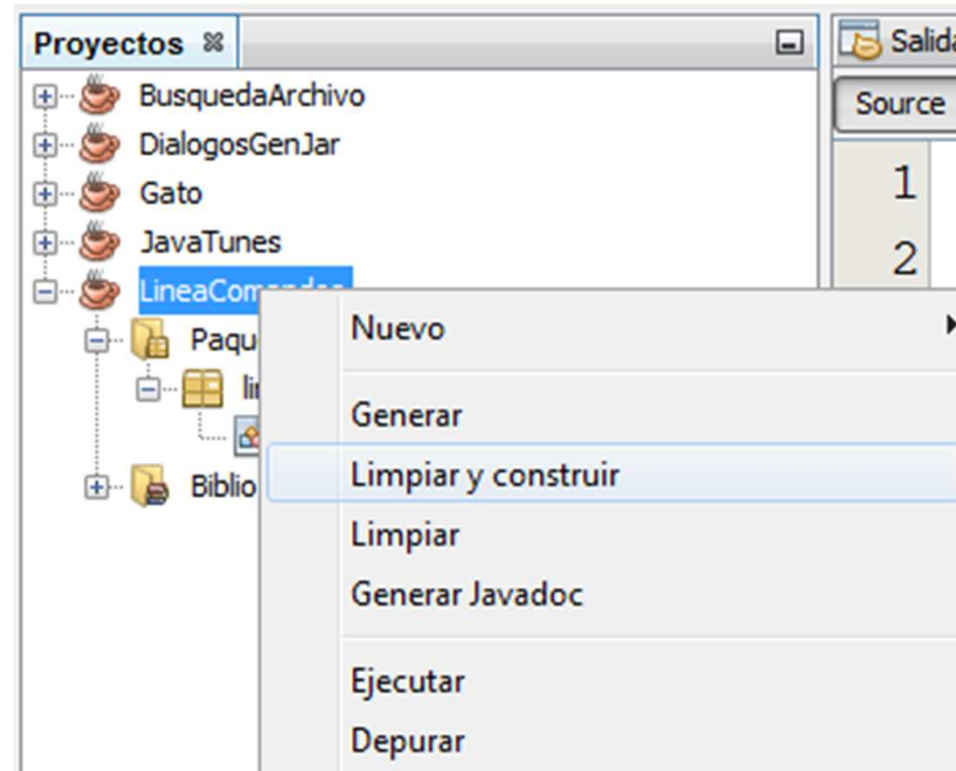
C:\Users\yhavhe>java -version
java version "1.7.0_03"
Java(TM) SE Runtime Environment (build 1.7.0_03-b05)
Java HotSpot(TM) 64-Bit Server VM (build 22.1-b02, mixed mode)

C:\Users\yhavhe>
```

```
public class LineaComandos {  
  
    public static void main(String[] args) {  
  
        for (int i = 0; i<args.length; i++){  
            System.out.print ("Parametro ["+i+"]: ");  
            System.out.println (args[i]);  
        }  
    }  
}
```


Creamos el Jar

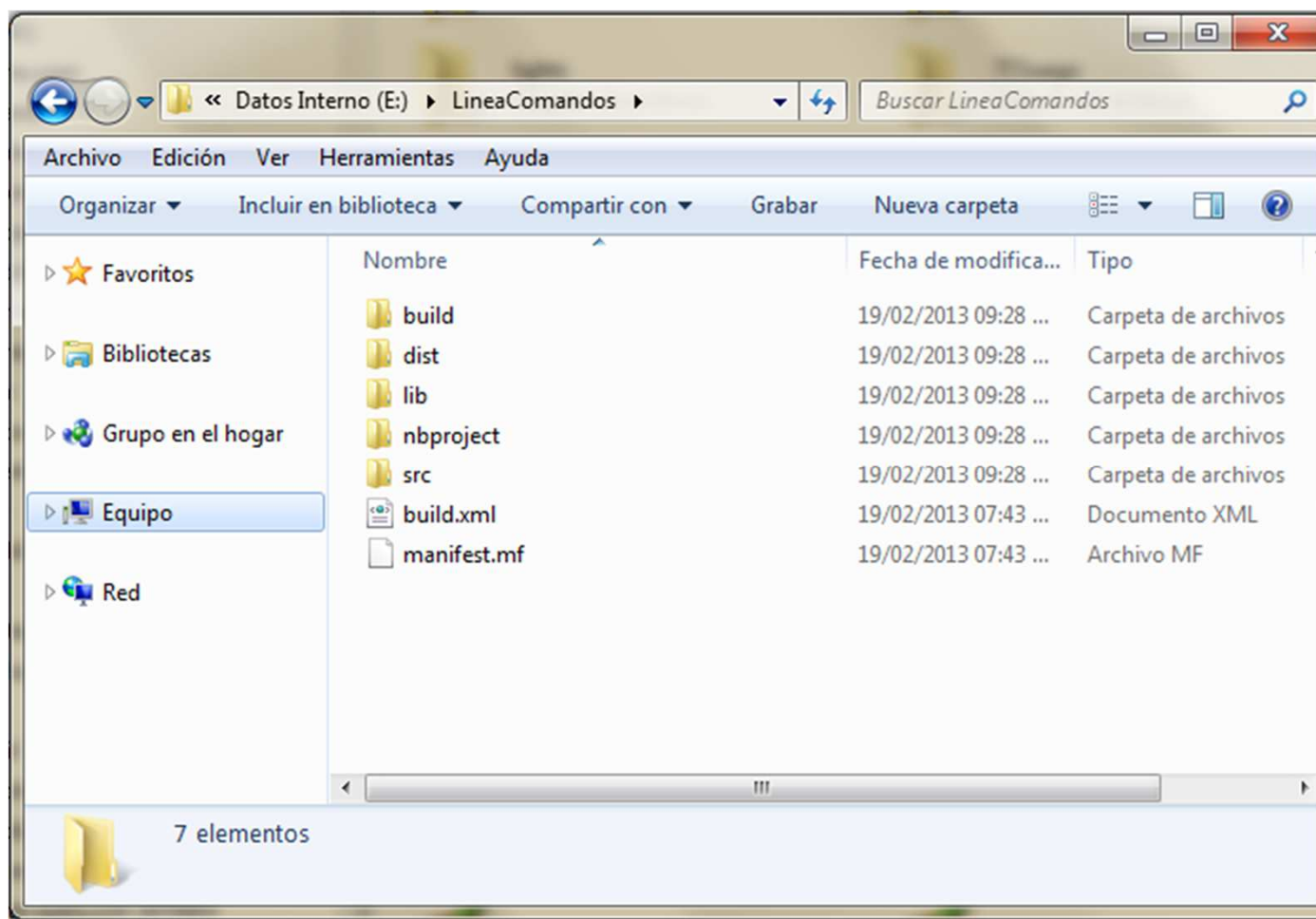
9



Abrimos un explorador

10

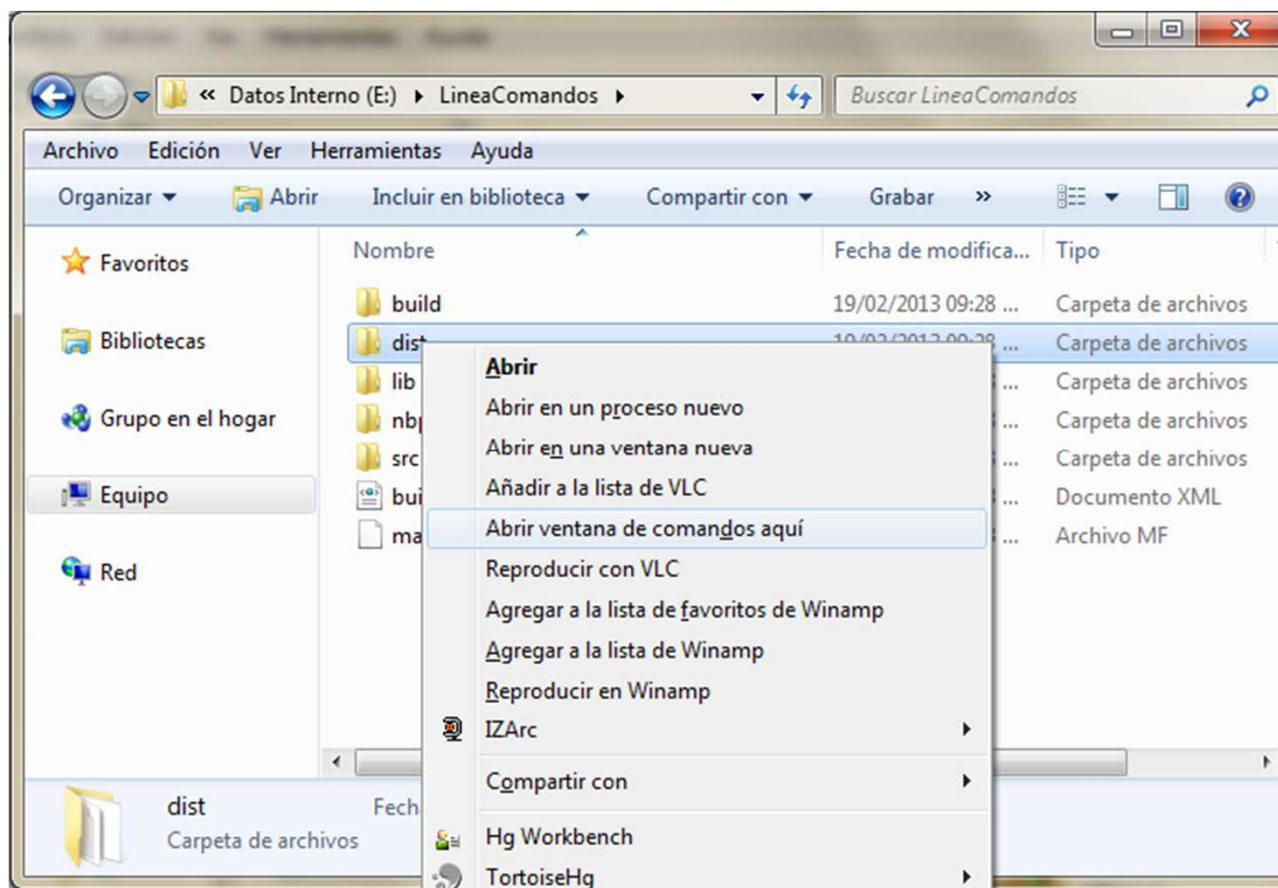
- Ir a la carpeta del proyecto



Shift + Click derecho

11

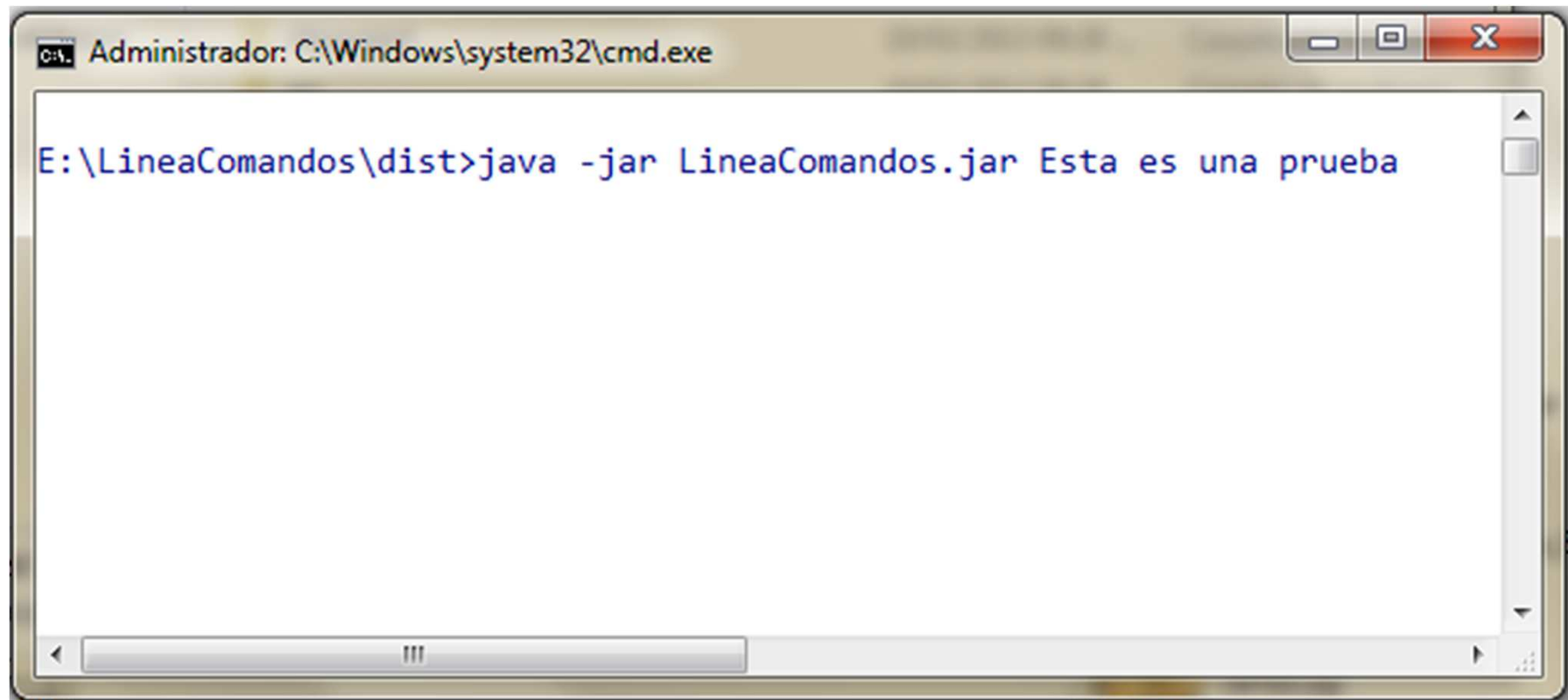
- Sobre la carpeta dist
 - "Abrir ventana de comandos aquí"



Ejecutando el programa

12

□ Escribimos:

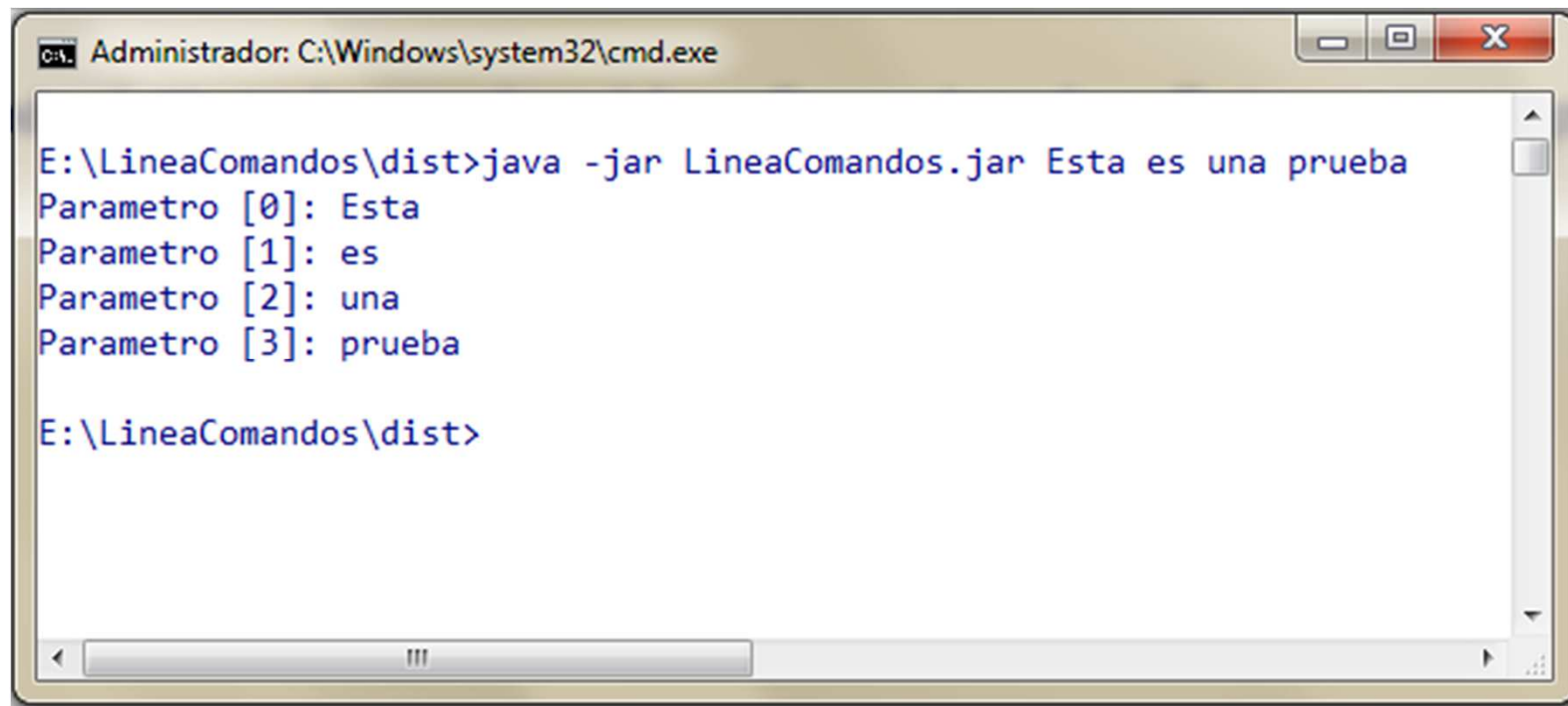


```
Administrador: C:\Windows\system32\cmd.exe

E:\LineaComandos\dist>java -jar LineaComandos.jar Esta es una prueba
```

Salida

13



A screenshot of a Windows command prompt window titled "Administrador: C:\Windows\system32\cmd.exe". The window shows the execution of a Java program. The command entered is `java -jar LineaComandos.jar Esta es una prueba`. The output shows the program parsing the arguments: `Parametro [0]: Esta`, `Parametro [1]: es`, `Parametro [2]: una`, and `Parametro [3]: prueba`. The prompt then returns to `E:\LineaComandos\dist>`.

```
Administrador: C:\Windows\system32\cmd.exe

E:\LineaComandos\dist>java -jar LineaComandos.jar Esta es una prueba
Parametro [0]: Esta
Parametro [1]: es
Parametro [2]: una
Parametro [3]: prueba

E:\LineaComandos\dist>
```

Revisando el código

14

```
public class LineaComandos {
```

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

```
        for (int i = 0; i < args.length; i++) {  
            System.out.print ("Parametro [" + i + "]: ");  
            System.out.println (args[i]);  
        }
```

```
    }
```


```
}
```

args

0	1	2	3
Esto	es	una	prueba

Revisando el código

15

```
public class LineaComandos {  
  
    public static void main(String[] args) {  
  
         for (int i = 0; i < args.length; i++) {  
            System.out.print ("Parametro [" + i + "]: ");  
            System.out.println (args[i]);  
        }  
    }  
}
```

i	args			
0	0	1	2	3
	Esto	es	una	prueba

Revisando el código

16

```
public class LineaComandos {  
  
    public static void main(String[] args) {  
  
        for (int i = 0; i<args.length; i++){  
            System.out.print ("Parametro ["+i+"]: ");  
            System.out.println (args[i]);  
        }  
    }  
}
```

Parametro [0]:

i

0

args

0	1	2	3
Esto	es	una	prueba

Revisando el código

17

```
public class LineaComandos {  
  
    public static void main(String[] args) {  
  
        for (int i = 0; i<args.length; i++){  
            System.out.print ("Parametro ["+i+"]: ");  
            System.out.println (args[i]);  
        }  
    }  
}
```

Parametro [0]: Esto

i	args			
0	0	1	2	3
	Esto	es	una	prueba



Revisando el código

18

```
public class LineaComandos {  
  
    public static void main(String[] args) {  
  
        → for (int i = 0; i<args.length; i++){  
            System.out.print ("Parametro ["+i+"]: ");  
            System.out.println (args[i]);  
        }  
    }  
}
```

Parametro [0]: Esto

i

1

args

0	1	2	3
Esto	es	una	prueba

Revisando el código

19

```
public class LineaComandos {  
  
    public static void main(String[] args) {  
  
        for (int i = 0; i<args.length; i++){  
            System.out.print ("Parametro ["+i+"]: ");  
            System.out.println (args[i]);  
        }  
    }  
}
```

Parametro [0]: Esto
Parametro [1]:

i
1

args			
0	1	2	3
Esto	es	una	prueba

Revisando el código

20

```
public class LineaComandos {  
  
    public static void main(String[] args) {  
  
        for (int i = 0; i<args.length; i++){  
            System.out.print ("Parametro ["+i+"]: ");  
            System.out.println (args[i]);  
        }  
    }  
}
```

Parametro [0]: Esto
Parametro [1]: es

i
1

args			
0	1	2	3
Esto	es	una	prueba



Revisando el código

21

```
public class LineaComandos {  
  
    public static void main(String[] args) {  
  
        → for (int i = 0; i<args.length; i++){  
            System.out.print ("Parametro ["+i+"]: ");  
            System.out.println (args[i]);  
        }  
    }  
}
```

Parametro [0]: Esto
Parametro [1]: es

i
2

args			
0	1	2	3
Esto	es	una	prueba

Revisando el código

22

```
public class LineaComandos {  
  
    public static void main(String[] args) {  
  
        for (int i = 0; i<args.length; i++){  
            System.out.print ("Parametro ["+i+"]: ");  
            System.out.println (args[i]);  
        }  
    }  
}
```


Parametro [0]: Esto
Parametro [1]: es
Parametro [2]:

i
2

args			
0	1	2	3
Esto	es	una	prueba

Revisando el código

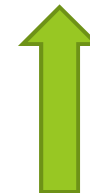
23

```
public class LineaComandos {  
  
    public static void main(String[] args) {  
  
        for (int i = 0; i<args.length; i++){  
            System.out.print ("Parametro ["+i+"]: ");  
            System.out.println (args[i]);  
        }  
    }  
}
```

Parametro [0]: Esto
Parametro [1]: es
Parametro [2]: una

i
2

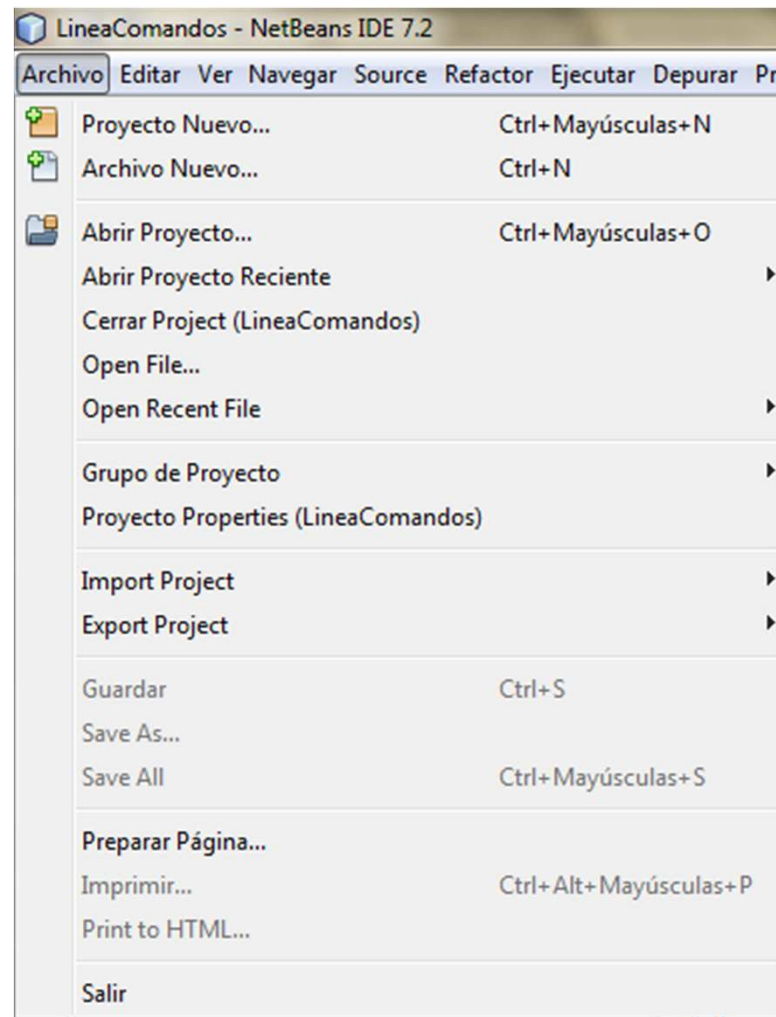
args			
0	1	2	3
Esto	es	una	prueba



Ejecutando desde netbeans

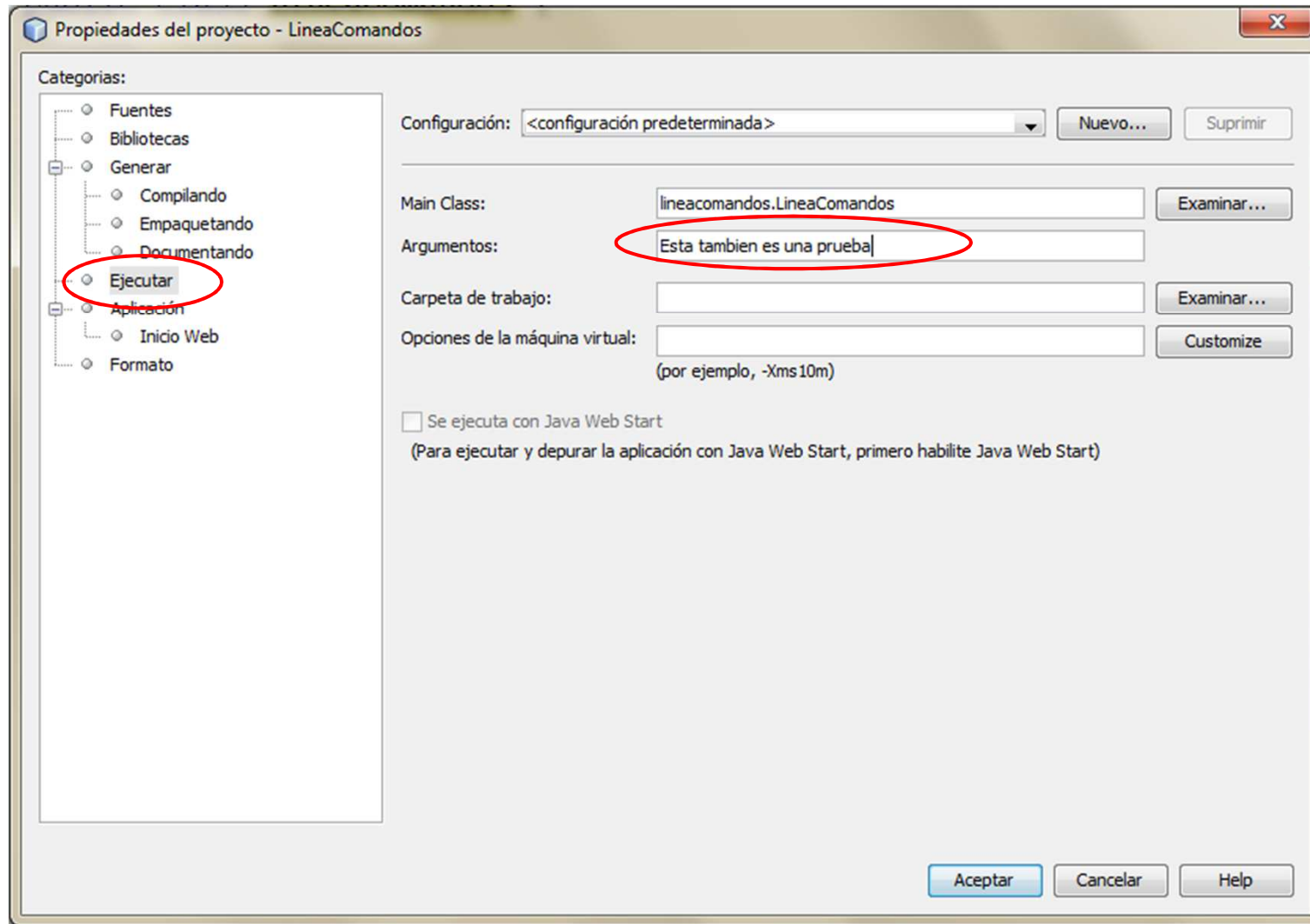
24

- Abrimos las propiedades del proyecto



Agregando parámetros

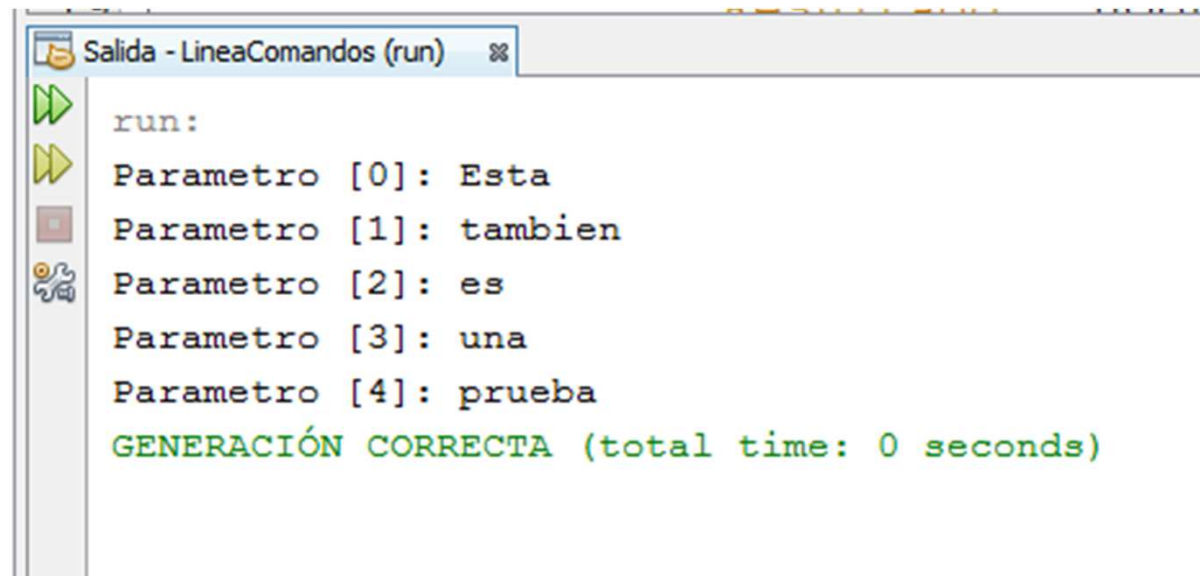
25



Ejecutando desde Netbeans

26

- Ejecutar como proyecto

A screenshot of the NetBeans console window. The title bar reads 'Salida - LineaComandos (run)'. The output text is as follows:

```
run:
Parametro [0]: Esta
Parametro [1]: tambien
Parametro [2]: es
Parametro [3]: una
Parametro [4]: prueba
GENERACIÓN CORRECTA (total time: 0 seconds)
```

Ejercicio



27

- ❑ Crea un programa llamado **Suma** que reciba N números por la línea de comandos.
- ❑ El programa debe sumarlos y mostrar el resultado.
- ❑ Prueba el programa desde la consola del sistema operativo



```
Administrador: C:\Windows\system32\cmd.exe

E:\>java -jar Suma.jar 200 23 5 18
La suma es: 246

E:\>
```