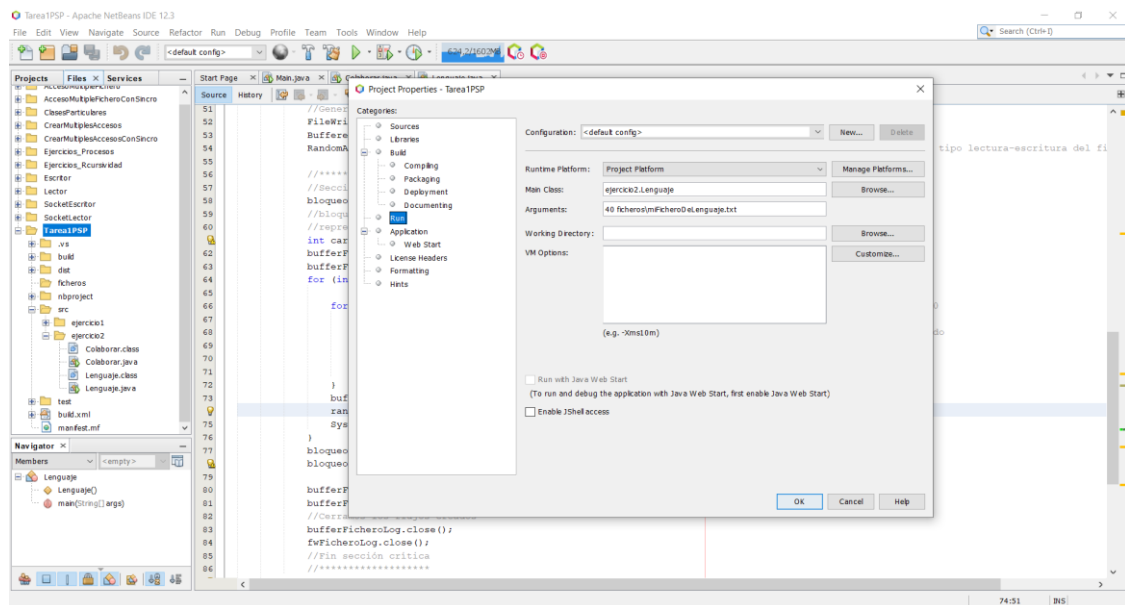


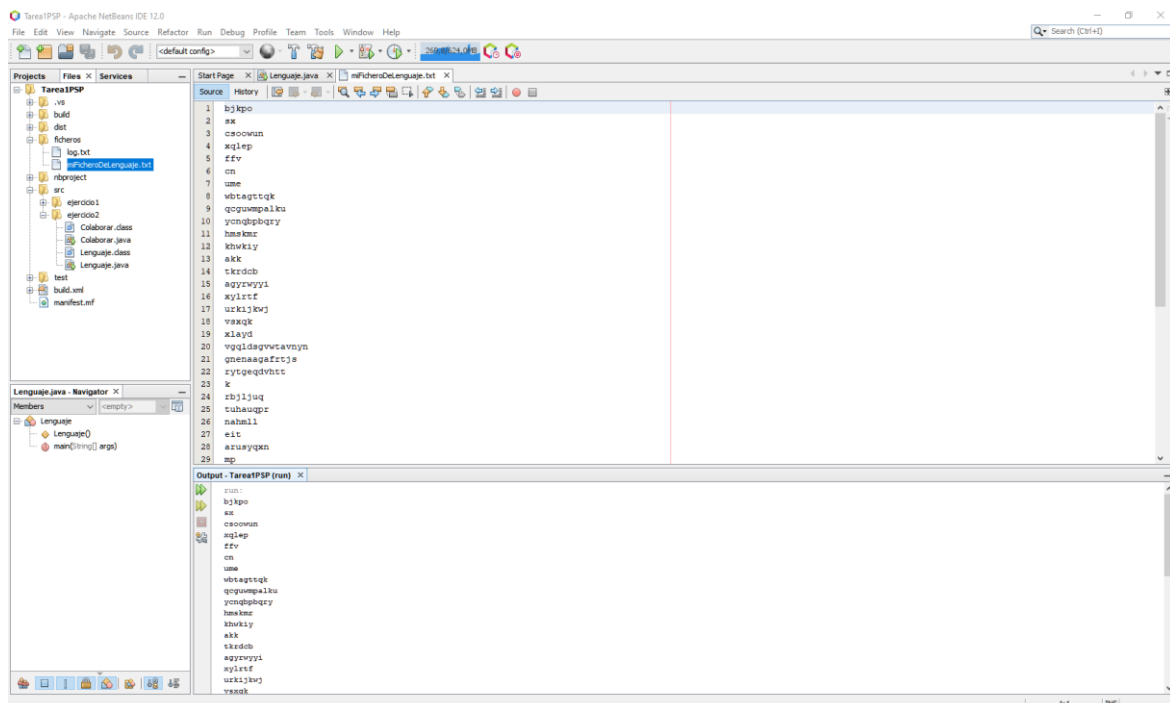
■ EJERCICIO 2 MANUAL DE PRUEBAS

- *Primera parte* Nos piden que la aplicación genere caracteres aleatorios formando palabras sin sentido escribiéndolas en un archivo en líneas distintas. El usuario decide cuantas palabras se generan y el nombre del archivo, es decir se lo pasamos como parámetros a la aplicación.

Esta sería la prueba estándar que nos sugiere la práctica:

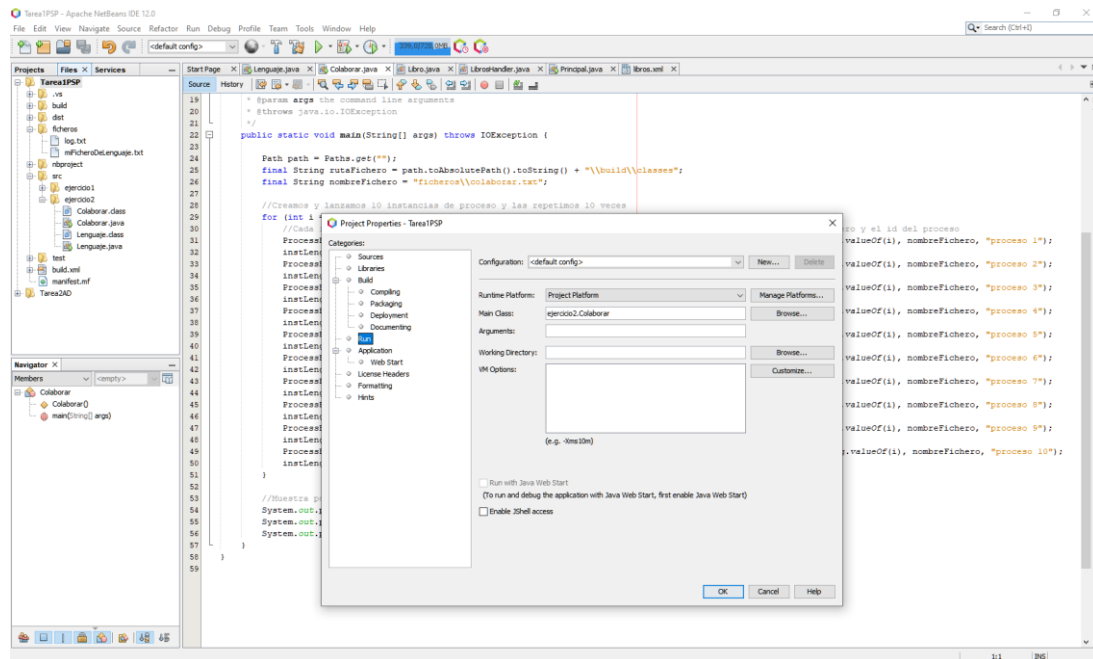


Pasamos como argumentos a la aplicación *40 ficheros\miFicheroDeLenguaje.txt* antes de ejecutarla

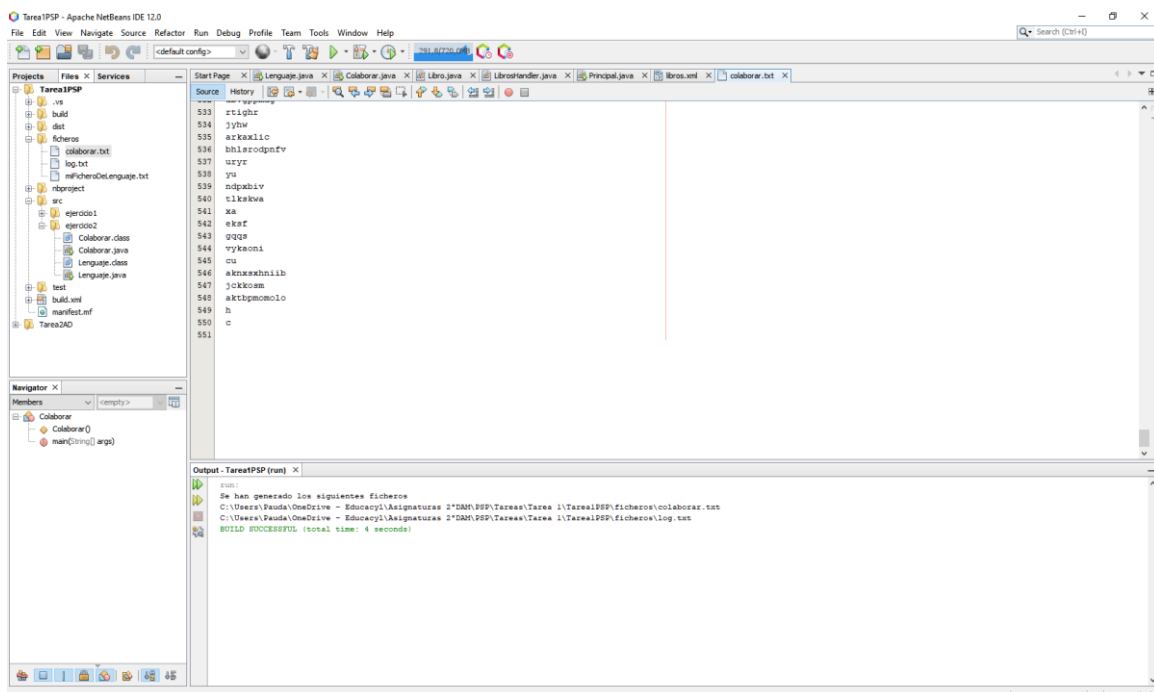


A la izquierda podemos ver que en la carpeta *ficheros* nos ha creado el archivo con el nombre que le hemos dado y a la dcha. podemos ver que nos ha escrito en el archivo las 40 palabras sin sentido real, a la vez que escribimos los caracteres en el archivo los imprimimos por pantalla para ver el resultado sin tener que abrir el fichero

- *Segunda parte* La aplicación creada llamada “Colaborar” crea 10 llamadas a la aplicación “Lenguaje” haciendo uso de la clase ProcessBuilder con la que podemos generar el proceso y mandar mediante comandos los argumentos que nos pide la aplicación.



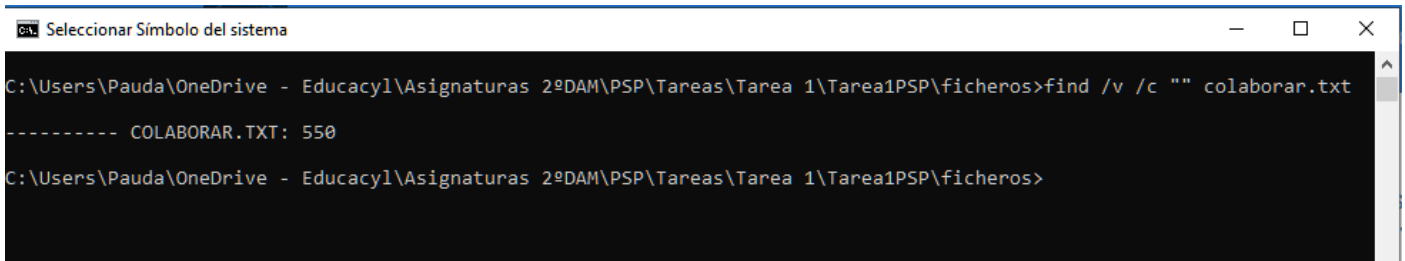
Dentro de las propiedades del proyecto seleccionamos la ejecución de la aplicación colaborar sin darle argumentos.



Al ejecutar podemos comprobar que nos ha creado un fichero (colaborar.txt) en el que los distintos procesos han ido escribiendo de manera controlada y secuencial hasta generar 550 palabras.

○ Tercera parte

Para poder comprobar mediante línea de comandos que nuestro fichero genera 550 líneas lo haremos de la siguiente manera:



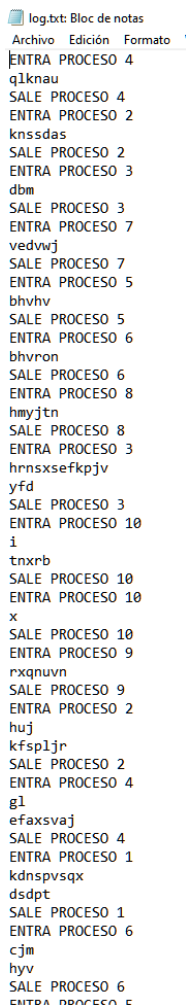
```

C:\Users\Pauda\OneDrive - Educacyl\Asignaturas 2ºDAM\PSP\Tareas\Tarea 1\Tarea1PSP\ficheros>find /v /c "" colaborar.txt
----- COLABORAR.TXT: 550
C:\Users\Pauda\OneDrive - Educacyl\Asignaturas 2ºDAM\PSP\Tareas\Tarea 1\Tarea1PSP\ficheros>

```

Utilizando este comando `find /v /c "" colaborar.txt` en Windows nos dará el como resultado el número de líneas que componen el fichero.

Otra prueba importante es la de saber si todos los procesos están escribiendo en el mismo fichero y si el bloqueo que hemos creado para que se sincronicen y entren uno detrás de otro funcionan bien. Para ello en paralelo a la escritura del fichero principal se va a ir generando un fichero al que hemos llamado log.txt en el que se escribirá cual es el número del proceso que entra a escribir, que es lo que escribe y cuando termina la escritura y comienza el siguiente:



```

log.txt: Bloc de notas
Archivo Edición Formato
ENTRA PROCESO 4
q1knau
SALE PROCESO 4
ENTRA PROCESO 2
knssdas
SALE PROCESO 2
ENTRA PROCESO 3
dbm
SALE PROCESO 3
ENTRA PROCESO 7
vedvwj
SALE PROCESO 7
ENTRA PROCESO 5
bhvhv
SALE PROCESO 5
ENTRA PROCESO 6
bhvron
SALE PROCESO 6
ENTRA PROCESO 8
hmyjtn
SALE PROCESO 8
ENTRA PROCESO 3
hrnsxsefkjv
yfd
SALE PROCESO 3
ENTRA PROCESO 10
i
tnxrb
SALE PROCESO 10
ENTRA PROCESO 10
x
SALE PROCESO 10
ENTRA PROCESO 9
rxqnuvn
SALE PROCESO 9
ENTRA PROCESO 2
huj
kfsp1jr
SALE PROCESO 2
ENTRA PROCESO 4
gl
efaxsvaj
SALE PROCESO 4
ENTRA PROCESO 1
kdnspsvx
dsdpt
SALE PROCESO 1
ENTRA PROCESO 6
cjm
hyv
SALE PROCESO 6
ENTRA PROCESO 5

```

Aquí podemos ver un pequeño ejemplo de como quedaría el fichero log.txt, en mayúsculas tenemos el número del proceso y en minúsculas que es lo que escribe