**EJERCICIO 02**

Comienzo creando dos proyectos, colaborar y lenguaje, con sus respectivas clases con el mismo nombre en un paquete main.

Para la clase lenguaje, lo primero es asegurar que sólo entran 2 parámetros al main por args, por lo que sí es distinto de 2, muestro error y cierro.

Acto seguido asigno los argumentos del main a las variables numeroPalabras y nombreFichero.

Creo un objeto random para hacer números aleatorios y creo un bucle dentro de otro, el primero genera un número aleatorio de letras por palabra y el segundo bucle crea letras aleatorias basándose en el código ascii de ‘a’ + el número aleatorio entre 1 y 26) siendo 1 = a y 26 = z.

Por último, con FileWriter escribo la palabra y un salto de línea en el fichero.

A continuación voy a la clase Colaborar y hago la misma condición que a Lenguaje, esta vez si el argumento es distinto de 1, muestro error y cierro.

Con un bucle for de 1 a 10, ejecuto el comando “java –jar lenguaje.jar “ + (i \* 10) + “ “ args[0] (siendo args[0] el nombre del archivo). Para que se ejecute 10 veces, siendo la primera vez 10, la segunda 20 y así sucesivamente hasta tener 550 líneas.

Finalmente, ejecuto el String comando con un Runtime.getRutime().exec(), que aunque me indica que esta deprecated, sigue funcionando. Aunque como alternativa se podría hacer con un ProcessBuilder pb, pb.command(), pb.start(), como inico en los comentario del código.

Genero los Javadocs con “Generate Javadoc” para cada proyecto y genero los .jar con “Clean and Build” para cada proyecto, y comprobamos que funcione llamando al colaborar con el siguiente comando en el terminal de Windows:  
java -jar colaborar.jar prueba.txt

