****

***Activad UT7.1***

***Programación***

*Alfonso García Jorge*

*1ºDAWNA*

**ACTIVIDAD 7.1**

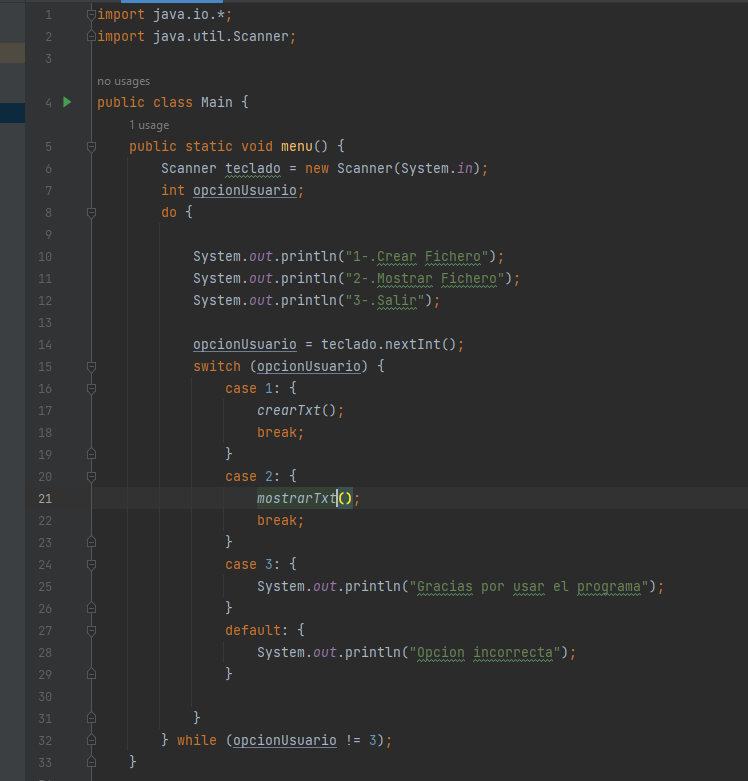
1.- Crear un menú con las siguientes opciones

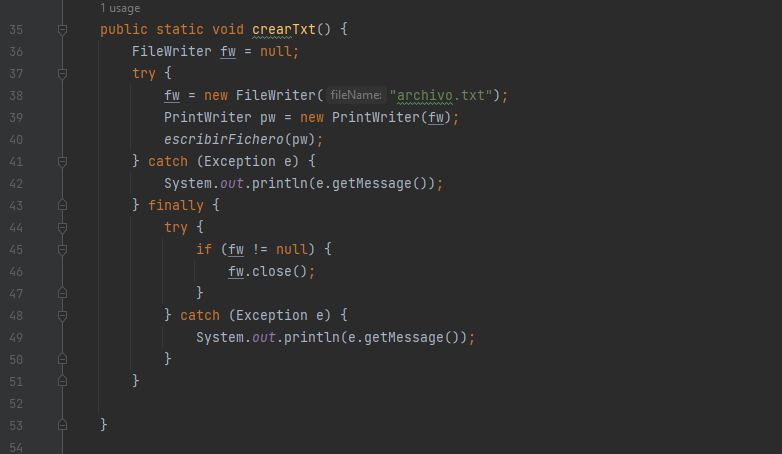
1. Crear un fichero texto (con el nombre que tú quieras) en el que indiques en cada línea separados por un espacio: (para esto podrás pedir por teclado)

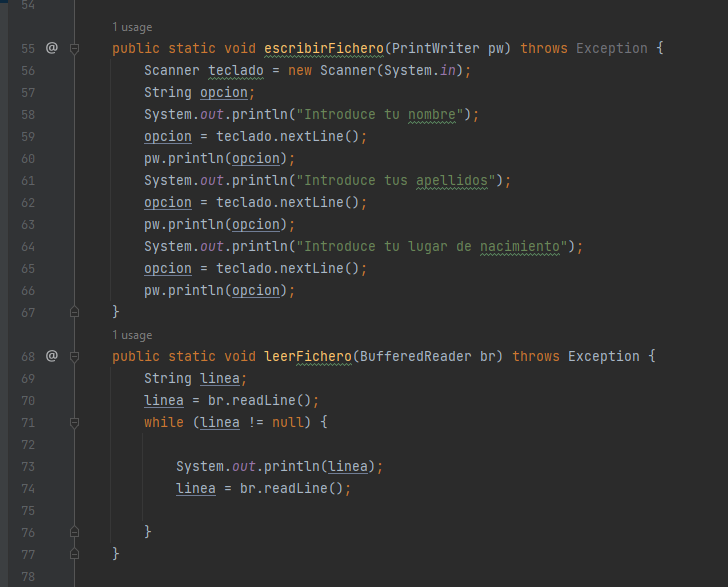
* Tu Nombre.
* Tu Edad.
* Tu Ciudad de Nacimiento.

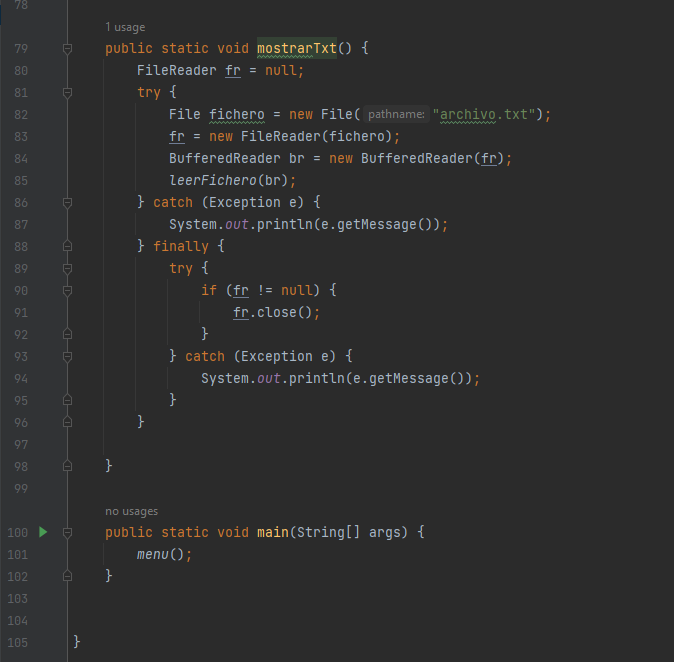
2. Mostrar por pantalla el contenido del fichero de texto creado.

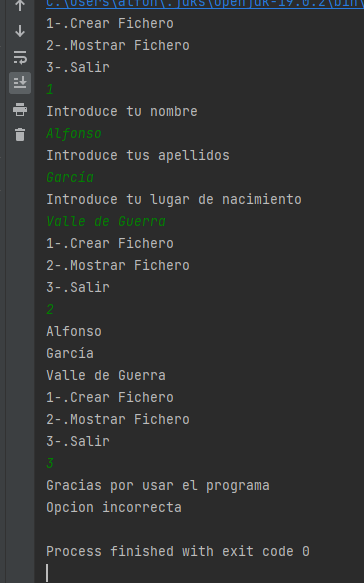
3. Salir del Programa.





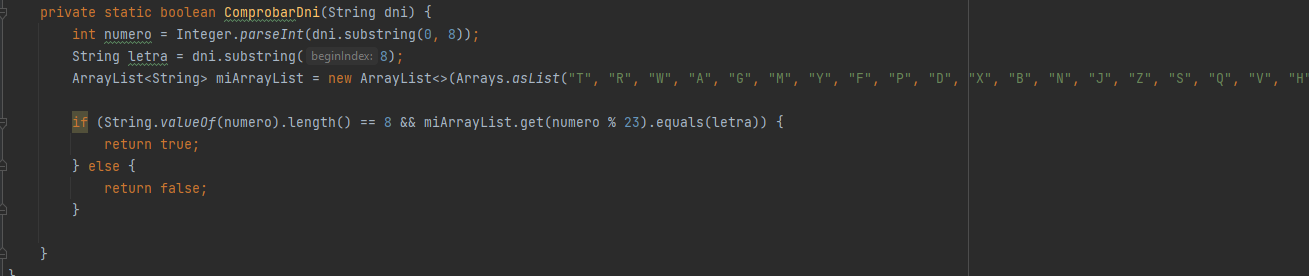






2.- Realiza un programa en JAVA en el que le pidas al usuario su DNI y:

* En caso de que el DNI sea correcto lo introducirás en un fichero llamado dni.txt.
* En caso de que el DNI sea incorrecto avisarás al usuario del error y no lo introducirás en el fichero.
* Cuando vuelvas a ejecutar el programa no se sobrescribirá el fichero, sino que se seguirán añadiendo DNIs validos al final del fichero dni.txt.
* Hay que calcular la letra del DNI usando el método matemático que corresponde.



3.- Realiza un programa en JAVA en el que muestres un menú que te permita 3 opciones:

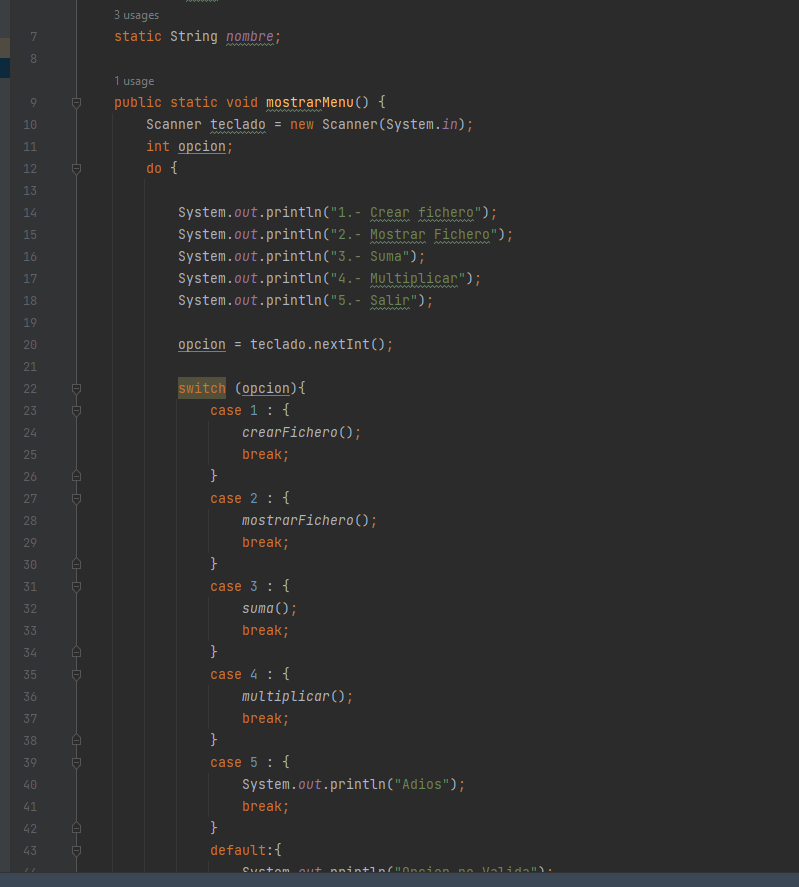
1. Volcado de un array con los 100 primeros números pares a un fichero de texto. El nombre del fichero lo elegirá el usuario.

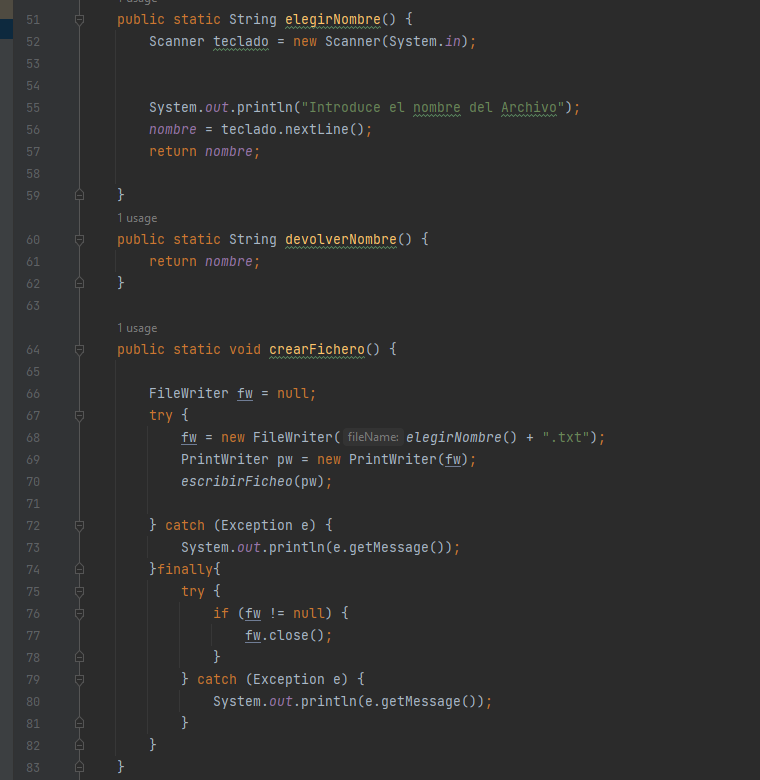
2. Mostrar por pantalla el contenido del fichero de texto creado.

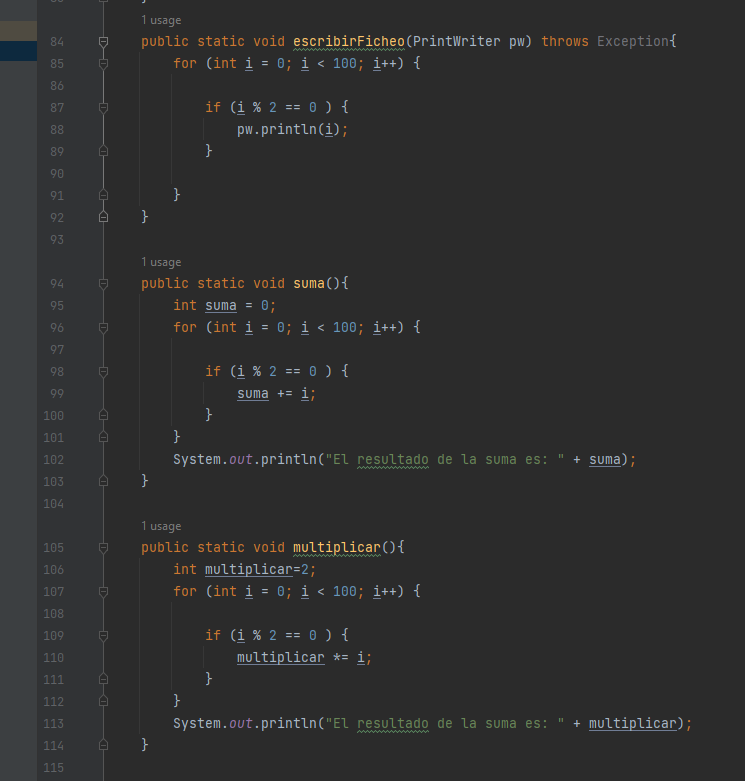
3. Calcular la suma de dicho fichero

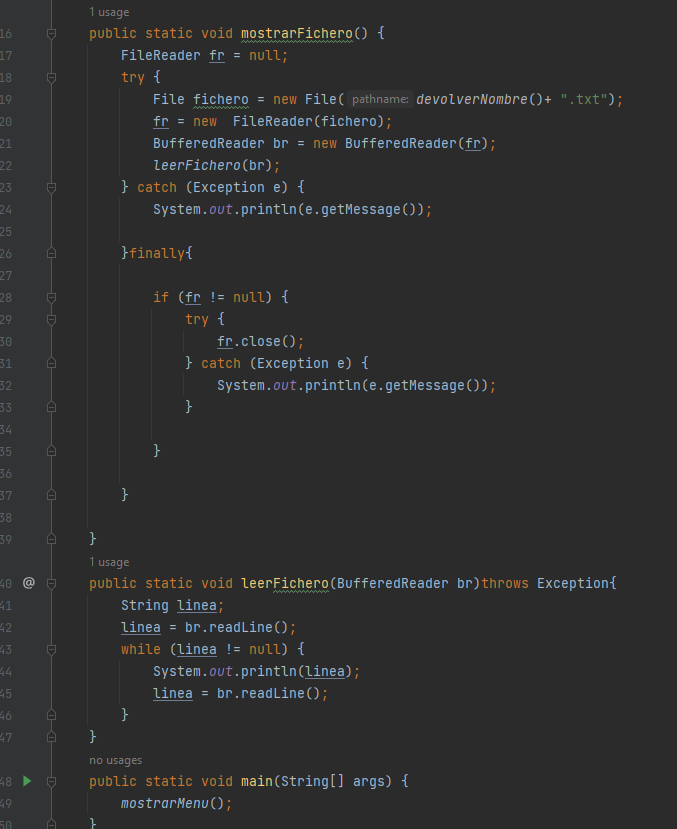
4. Calcular la multiplicación de dicho fichero

5. Salir del Programa.









4.- Se tiene un archivo de caracteres de nombre “SALAS.DAT” (Crear dicho archivo con al menos 20 líneas). Escribir un programa para crear el archivo “SALAS.BIN” con el contenido del primer archivo, pero en modo binario.

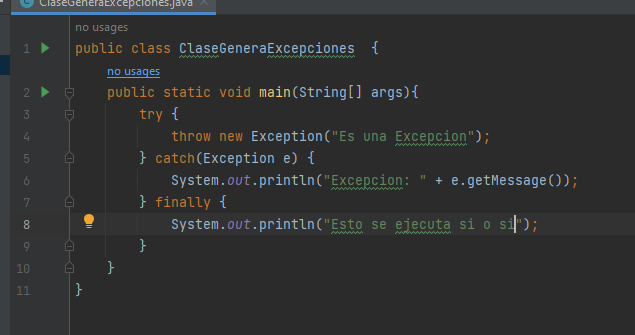
5.- Construir un directorio de amigos (Usando clases y una colección). Considérese que la información más relevante para representar a un amigo es: nombre, teléfono, correo-e y fecha de cumpleaños.

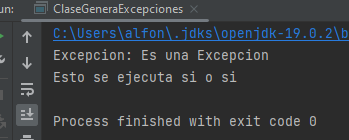
Utilizar objetos serializables para escribir y leer de archivo.

6.- El fichero calificaciones.csv contiene las calificaciones de un curso. Durante el curso se realizaron dos exámenes parciales de teoría y un examen de prácticas.

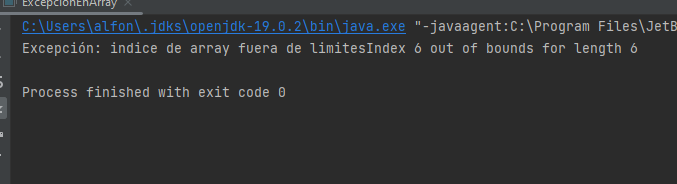
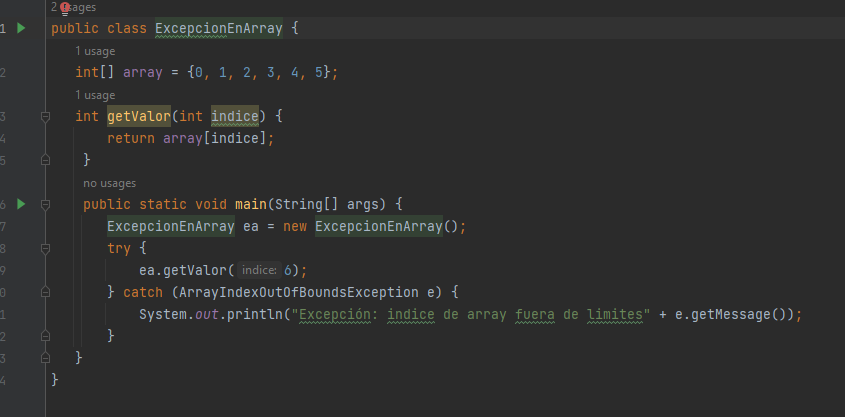
* Un método que reciba el fichero de calificaciones y devuelva un HashTree, donde cada elemento contiene como clave los apellidos y el nombre de cada alumno y de valor la asistencia a clase sin el signo de %.
* Calcula la media de los 2 parciales de los alumnos creando un fichero nuevo donde aparezcan los apellidos y nombre en una columna y la media en otra.
* Calcula la nota media del Ordinario 1 y Ordinario2 de cada alumno creando un fichero nuevo donde aparezcan los apellidos y nombre en una columna y la media de los ordinarios y en otra.
* Imprime por pantalla la media de asistencia del alumnado.
* Imprime por pantalla el número de alumnos que han sacado más de un 5 en el parcial 1, parcial 2 y prácticas. Considera que el alumno que no tenga una nota, su valor es 0.

7.- Cree una clase con un método main() que genere un objeto de la clase Exception dentro de un bloque try. Proporcione al constructor de Exception un argumento String. Capture la excepción dentro de una cláusula catch e imprima el argumento String. Añada una cláusula finally e imprima un mensaje para demostrar que pasó por allí.





8.- Escriba código para generar y capturar una excepción ArrayIndexOutOfBoundsException (Índice de matriz fuera de límites).



9.- Cree su propia clase de excepción utilizando la palabra clave extends. Escriba un constructor para dicha clase que tome un argumento String y lo almacene dentro del objeto como una referencia de tipo String. Escriba un método que muestre la cadena de caracteres almacenada. Cree una cláusula try-catch para probar la nueva excepción.



