

Funcionalidad

- Las opciones del menú funcionan y el usuario o usuaria puede elegir el nombre de los ficheros que quiere usar.
- En el caso de los ficheros de lectura, se pedirán nombres hasta que el nombre introducido sea válido.

Para ello se ha usado el método `conectarFichero()`.

- Se utilizará correctamente la sentencia `try-catch-finally` para evitar que el programa falle por una excepción producida cuando se intenta ver un "Mad Lib". Se ha usado en el método `verMadLib()`.

```
public static void VerMadlib(Scanner teclado) throws
FileNotFoundException {
    Scanner fichero = null;
    try {
        fichero = conectarFichero(teclado);
    } catch (FileNotFoundException excepcion) {
        System.out.println("Excepción");
    } catch (Exception excepcion) {
        System.out.println("Otras excepción");
        System.out.println(excepcion.getMessage());
    }
    while(fichero.hasNextLine()) {
        System.out.println(fichero.nextLine());
    }
    if (fichero != null) {
        fichero.close();
    }
}
```

- El programa está bien estructurado y no hay redundancias.
- Se utilizan 4 métodos con paso de parámetros y al menos uno de ellos usa la sentencia `return`. Además a uno de ellos se le pasará un objeto como parámetro y otro devolverá un objeto como parámetro.

1. Se han creado 5 métodos con paso de parámetros

```
public static String seleccionarOpcion(Scanner leerTeclado){}
public static void VerMadlib(Scanner teclado) throws FileNotFoundException
{}
public static void procesarSalida(Scanner entrada, Scanner teclado) {}
```

2. Por lo menos uno de ellos devuelve un objeto

```
public static Scanner conectarFichero(Scanner teclado) {}
```

3. Por lo menos uno de ellos recibe un objeto como parámetro

```
public static void procederSeleccion(Scanner teclado, String eleccion){}
```

Nota: 6.5

Comentarios

- Están bien escritos, en los lugares apropiados y ayudan a entender el funcionamiento del programa.

Nota: 1.5

Legibilidad

- El código está bien organizado, sigue la guía de estilo y es fácil de leer.

Nota: 1,5

Nota de la autoevaluación 9,5

Programa

```
import java.io.*;
import java.util.*;
public class MadLibs_V01 {
    public static final String CREAR = "c";
    public static final String VER = "v";
    public static final String SALIR = "s";
    public static void main(String[] args) throws FileNotFoundException {
        Scanner teclado = new Scanner(System.in);
        String seleccionUsuario = "";
        introduccion();
        do{//mientras que la entrada por teclado no sea "s" seguirá mostrando
            seleccionUsuario = seleccionarOpcion(teclado);
            procederSeleccion(teclado, seleccionUsuario);
            System.out.println();
        }while(!seleccionUsuario.equals("s"));
    }
    /***** MÉTODOS *****/
    //Método que muestra la introducción al juego
    public static void introduccion(){
        System.out.println("Bienvenidos y bienvenidas al juego de los cuentos locos.");
        System.out.println("El programa te pedirá que introduzcas una serie de palabras");
        System.out.println("que se utilizarán para completar una historia.");
    }
}
```

```

        System.out.println("El resultado se guardará en un fichero.");
        System.out.println("Puedes leer esas historias siempre que quieras.");
        System.out.println();
    }
    //Método que muestra el menu
    public static void menu(){
        System.out.println("***** MENU *****");
        System.out.println("(C)rear un \"Mad Lib\"");
        System.out.println("(V)er un \"Mad Lib\"");
        System.out.println("(S)alir");
        System.out.println("Elija su opción: ");
    }
    //Método para seleccionar la opción. Se le pasa el objeto Scanner para leer por
    teclado la opcion c, v o s
    //que son los valores de las constantes que se han creado para las opciones.
    public static String seleccionarOpcion(Scanner leerTeclado) {
        menu();
        String eleccion = leerTeclado.next();
        eleccion = eleccion.toLowerCase();
        while(!eleccion.equals(CREAR) && !eleccion.equals(VER) && !eleccion.equals(SALIR))
    {
        menu();
        eleccion = leerTeclado.next();
        eleccion = eleccion.toLowerCase();
    }
    return eleccion;
}
    //Método para trabajar con la selección, se le pasa el objeto Scanner para leer por
    teclado y un variable con la elección
    public static void procederSeleccion(Scanner teclado, String eleccion) throws
FileNotFoundException {
        if(eleccion.equals(CREAR)){
            System.out.println("Crear un cuento:");
            CrearMadLib(teclado);
        }
        else if(eleccion.equals(VER)){
            System.out.println("Ver un cuento:");
            VerMadlib(teclado);
        }else{
            System.out.print("Agur");
        }
    }
    //antes se devolvía la eleccion
}
    //Método para crear el Mad Lib, se le pasa un Objeto Scanner para leer por teclado
    public static void CrearMadLib(Scanner teclado) throws FileNotFoundException {
        procesarSalida(conectarFichero(teclado), teclado);
    }
    //Método para ver en pantalla el Mad Lib, se le pasa un Objeto Scanner para leer por
    teclado
    public static void VerMadlib(Scanner teclado) throws FileNotFoundException {
        Scanner fichero = null;
        try {
            fichero = conectarFichero(teclado);
        } catch (FileNotFoundException excepcion) {
            System.out.println("Excepción");
        } catch (Exception excepcion) {
            System.out.println("Otras excepción");
            System.out.println(excepcion.getMessage());
        }
        while(fichero.hasNextLine()) {
            System.out.println(fichero.nextLine());
        }
    }
}

```

```

    }
    if (fichero != null) {
        fichero.close();
    }
}

//Método para conectar con el fichero Se le pasa el objeto Scanner para leer por
teclado y devuelve un objeto File.
public static Scanner conectarFichero(Scanner teclado) throws FileNotFoundException {
    System.out.println("Nombre del fichero que quieres leer: ");
    String nombre = teclado.next();
    File ficheroEntrada = new File(nombre);
    //Hasta que el fichero se pueda leer lo pedirá
    while(!ficheroEntrada.canRead()){
        System.out.println("Fichero no encontrado. Inténtalo otra vez ");
        System.out.println("Nombre del fichero: ");
        nombre = teclado.next();
        ficheroEntrada = new File(nombre);
    }
    return new Scanner(new File(nombre));
}

//Método para escribir en el fichero de salida.Se le pasan 2 objetos Scanner
//Uno para leer el fichero y otro para leer el teclado
public static void procesarSalida(Scanner entrada, Scanner teclado) throws
FileNotFoundException{
    System.out.println("Nombre del fichero de salida: ");
    String nombreSalida = teclado.next();
    File ficheroSalida = new File(nombreSalida);
    PrintStream escribirFicheroSalida = new PrintStream(ficheroSalida);
    //mientras que tenga para leer líneas las leerá
    while(entrada.hasNextLine()){
        String linea = entrada.nextLine();
        Scanner leerLinea = new Scanner(linea);
        //mientras que tenga para leer palabras las leerá
        while(leerLinea.hasNext()){
            String palabra = leerLinea.next();
            //Si la palabra no empieza por < y termina por >
            if(!palabra.startsWith("<") && !palabra.endsWith(">")){
                //escribirá en el fichero de salida con un espacio en blanco
                escribirFicheroSalida.print(palabra + " ");
                //si empieza por < y termina por >
            }else{
                palabra = palabra.replace("-", " "); // si tiene un guión lo reemplazará
                palabra = palabra.replace("<", ""); //eliminará <
                palabra = palabra.replace(">", ""); //eliminará >
                System.out.println(palabra + ": "); //mostrara la palabra y le pondra 2
                puntos, para que el usuario responda
                String pedirPalabra = teclado.next(); //guardará la palabra del usuario
                escribirFicheroSalida.print(pedirPalabra + " "); // y escribirá en el
                fichero de salida
            }
        }
        escribirFicheroSalida.println();
    }
    System.out.println("El cuento loco ha sido creado");
    escribirFicheroSalida.close();
    System.out.println();
}
}

```