

MANUAL TECNICO

Sopa de letras

Manual técnico

Objetivo

Entender el funcionamiento del programa informático explicando sus funcionalidades a usuarios avanzados que tengan un conocimiento pragmático del lenguaje con el que se ha trabajado esta aplicación de consola.

Especificaciones

Requerimientos del sistema

- Sistema operativo (Windows, Linux , Ubuntu , IOS).
- Un procesador de 64bits o de 32bits.
- Instalado Java en el computador.
- Memoria RAM 1GB.
- Sin conexión a internet.
- Sin requerimientos de tarjeta gráfica.

Versión de Java

- 1.8.0_231

Versión jre

- 1.8.0_231-b11

IDE utilizado

- Eclipse (4.10).

Lenguaje utilizado

- Java nativo

En este apartado se creó la clase principal “practicanueva” donde se creó un método llamado “main”, este método contiene toda la lógica del programa.

Al inicio de la clase se colocó un println, este println nos muestra el título principal del programa.

A continuación del println se colocaron las variables globales a utilizar en el programa estas variables son de diferentes tipos de datos string y int.

Lo descrito se puede observar en la imagen.

```
package practicanueva;
import java.util.Scanner;
public class practicanueva {
    public static void main(String[] args) {
        System.out.println("SOPA DE LETRAS");
        System.out.println("*****");
        String menu = "";
        String palabrasEncontradas="0";
        String palabrasNoEncontradas = "0";
        int menuParseado=0;
        String menuSegundo = "";
        String puntosfinal="";
        int menuSegundoparseado=0;
        String tercerMenu = "";
        String nombresGuardados = "";
        String nombresGuardadosGanados = "";
        String palabrasGuardadas = "";
        String todo="";
        String puntosFinal="";
        String arreglopalabras[]=new String[1];
        String nombre = "";
        int contador2=0;
        int parseado =0;
        int estadisticasParcear = 0;
```

Se creó un do while para hacer una regresión al menú secundario.

Se creó una variable de tipo Scanner para poder instanciar un objeto de tipo escáner.

Se utilizó System.out.println para mostrar el menú secundario en la consola

Se creó una variable String palabrasNoGuardadas vacía donde se usara mas adelante-

Se utilizó un método del objeto Scanner llamado next() donde este método hace que se pueda escribir en consola y retornar el valor colocado.

Se utilizó un método llamado parseInt para modificar el tipo de dato de la variable menú donde la variable menú es de tipo String y se pasó a tipo Int con el método parseInt.

```
do{
    Scanner scanner = new Scanner(System.in);
    System.out.println("1)nueva partida 2)estadisticas 3)salir del juego");
    System.out.println("*****");
    System.out.println("ingrese el numero de la opcion que desea ingresar");
    String palabrasNoGuardadas="";
    menu = scanner.next();
    menuParseado= Integer.parseInt(menu);
    ...
```

Se creó un switch para colocar un caso por una opción del menú principal

El caso 1 hace referencia a la opción “nueva partida”.

Se crea un System.out.print preguntando cual es su nombre.

Luego de eso se creó otro System.out.print para preguntar el tamaño de la matriz.

Se creo un while() con la condición que la variable “par” sea menor a 10.

Entrando al ciclo se encuentra una variable “tamañoMatriz” obteniendo los datos puestos en consola.

Donde después se utiliza la variable “par” para cambiar de dato a la variable “tamaño matriz”.

se creo un if () else para hacer una restricción al usuario que tiene que ser igual o mayor a 5 el arreglo.

```
switch(menuParseado) {  
case 1:  
    System.out.println("¿cual es tu nombre?");  
    nombre = scanner.next();  
    System.out.println("Escoje el tamano del la matriz");  
    int par =0;  
    String tamañoMatriz="";  
    while(par<10){  
        tamañoMatriz = scanner.next();  
        par = Integer.parseInt(tamañoMatriz);  
        if(par<10){  
            System.out.println("tiene que ser igual o mayor a 5 el arreglo");  
        }else{  
        }  
    }  
}
```

Se creo un if para hacer una restriccion donde si par es menor a 9 saldra el mensaje “tiene que ser igual o mayor a 10 el arreglo”

En caso contrario se ejecutara un do while donde este do while ejecuta la parte del segundo menu.

Luego de esto se crea un switch evaluando la variable “menu SegundoParceado”

Donde el primer caso representaria “menu palabras”

Entrando al caso 1 se encuentra un do while donde contiene las opciones del menu de palabras

Usando la variable “tercerMenu” para retornar el valor ingresado por el usuario.

Luego se utiliza la variable “parseado” para cambiarle el tipo de dato a la variable “tercerMenu”

Se crea un switch para poder retornar al menu palabras.

Dentro de ese switch se imprime el mensaje “ingrese el numero de palabras”

Luego “ingrese la palabras”

```
if (par < 9) {
    System.out.println("tiene que ser igual o mayor a 10 el arreglo");
} else {
    do {
        System.out.println("1)menu palabras 2)jugar 3)salir de la partida");
        System.out.println("ingrese el numero de la opcion que desea ingresar");
        menuSegundo = scanner.next();
        menuSegundoParseado = Integer.parseInt(menuSegundo);
        switch (menuSegundoParseado) {
            case 1:
                do {
                    System.out.println("1)insertar 2)insertar todas 3)modificar 4)eliminar 5)salir");
                    System.out.println("ingrese el numero de la opcion que desea ingresar");
                    tercerMenu = scanner.next();
                    parseado = Integer.parseInt(tercerMenu);
                    switch (parseado) {
                        case 1:
                            System.out.println("ingrese el numero de palabras");
                            int numeroPalabras = scanner.nextInt();
                            System.out.println("ingrese las palabras");
                            int a = 0;
```

Se crea un while donde la condición es que la variable “a ” sea menor a la variable ”numero Palabras”

Luego se crea un if con la condición que las longitudes de las palabras sean mayor a 5 y que las longitudes de las palabras sean menores o iguales a 10.

Luego de entrar al if hay otro if donde la longitud de la palabra tiene que ser menor a la variable “par”.

A la variable palabras se le sumaran las palabras cada vez que recorra el while hasta que el while salga del ciclo.

La variable “contador2 ” aumentara en uno cada vez que el ciclo se repita.

En caso contrario del segundo if enviara un mensaje “la palabra tiene que ser menor a la matriz ”.

En el caso contrario del primer if enviara un mensaje “la palabra tiene que tener entre 5 y 10 caracteres”

Luego el caso 2 representa la opción “insertar todas” cuando se entra al caso se mostrara el mensaje “ingrese las palabras con un espacio de \$ por cada palabra”.

A la variable palabrasNoGuardadas se le sumaran los string que pongan en consola.

```
while(a<numeropalabras){  
    String palabras = scanner.next();  
    if (palabras.length()>=5&&palabras.length()<=10){  
        if (palabras.length()<par){  
            palabrasNoGuardadas+=palabras+"$";  
            palabrasGuardadas+=palabrasNoGuardadas;  
            ++contador2;  
            ++a;  
        }else{  
            System.out.println("la palabra tiene que ser menor a la matriz");  
        }  
    }else{  
        System.out.println("la palabra tiene que tener entre 5 y 10 caracteres");  
    }  
}  
break;  
case 2 :  
    System.out.println("ingrese las palabras con un espacio de $ por cada palabra");  
    palabrasNoGuardadas += scanner.next();  
    palabrasGuardadas += palabrasNoGuardadas;  
    break;
```

El caso 3 representa “insertar una palabra” cuando se ingresa ha caso hay una condicion if donde la condicion es palabrasNoGuardadas es igual a una cadena vacia.

En caso contrario se ejecuta un system.out.print con en mensaje “escriba la palabra que quiera modificar”. Utilizando scanner se obtiene la palabra modificada.

Luego se inicializa un if con la condicion palabrasNoGuardadas.contains(modificar)==true.

Si la condicion es verdadera saldra un mensaje “escriba la palabra nueva”

Luego se crea una variable palabrasNueva donde se agregaran las palabras nuevas.

Luego con el metodo replace se obtiene un remplazamiento en la variable “palabrasNoGuardadas”.

Luego saldra un mensaje “se ha modificado ”.

El caso 4 representa “eliminar” cuando se ingresa a el caso se ejecutara un if con una condicion, la condicion es palabrasGuardadas ==”” en caso contrario saldra un mensaje “escriba la palabra que quiere eliminar” luego se ejecutara un if con condicion

palabrasNoGuardadas. Contains (eliminar)==true donde si es verdadero eliminara la palabras en caso contrario si no existe la palabra saldria un mensaje “no se ha encontrado la palabra escrita”

```

case 3:
    if (palabrasNoGuardadas=="") {
    }else{
        System.out.println("escriba la palabra que quiere modificar");
        String modificar = scanner.next();
        if (palabrasNoGuardadas.contains(modificar)==true) {
            System.out.println("escriba la palabra nueva");
            String palabraNueva = scanner.next()+"$";
            palabrasNoGuardadas= palabrasNoGuardadas.replace(modificar, palabraNueva);
            palabrasGuardadas = palabrasNoGuardadas;
            System.out.println("se ha modificado");
            System.out.println(palabrasGuardadas);
        }else{
            System.out.println("no ha sido encontrada");
        }
    }
    break;
case 4 :
    if (palabrasGuardadas=="") {
    }
    else{
        System.out.println("escriba la palabra que quiere eliminar");
        String eliminar = scanner.next()+"$";
        if (palabrasNoGuardadas.contains(eliminar)==true) {
            palabrasNoGuardadas = palabrasNoGuardadas.replace(eliminar, "");
            palabrasGuardadas = palabrasNoGuardadas;
            System.out.println(palabrasGuardadas);
        }else{
            System.out.println("la palabra no fue encontrada");
        }
    }
    break;
}
}while (parseado != 5);

```

se ejecutara un if con una condicion que es la siguiente este if es una validacion por si no se han agregado palabra no comience el juego si no hay palabras aparecera el mensaje “no hay palabras agregadas” en caso contrario que si hayan palabras se ejecutara la matriz de dos dimensiones donde las dimensiones son el valor que puso el usuario al principio del programa este valor se guardo en la variable “tamañoMatrizparseada”, se crea un vector del avecedario.

Luego se ejecuta un for donde este for recorre la matriz principal pero antes de recorre se ejecuta otro for donde este recore la posicion en horizontal dentro del ultimo for se crea un numero random para colocar en la matriz luego se imprime la matriz ya con los valores aleatorios.

```

e 2 :
if (palabrasNoGuardadas=="") {
    System.out.println("no hay palabras agregadas");
}else{
    int tamañoMatrizparseada = Integer.parseInt(tamañoMatriz);
    String matrizPrincipal[][]= new String [tamañoMatrizparseada][tamañoMatrizparseada];
    String abc[] = {"a","b","c","d","e","f","g","h","i","j","k","l","m","n","ñ","o","p","q","r","s","t","u","v","w","x","y","z"," "};
    for(int c =0; c<matrizPrincipal.length;++c){
        for(int d =0; d<matrizPrincipal.length;++d){
            double random = Math.random()*26;
            int random2 = (int)random;
            matrizPrincipal[c][d]= abc[random2];
        }
    }
}

```

Se crea un while para las tres vidas que tiene el jugador el while sirve para hacer el ciclo tres veces hasta que ya no tenga vidas o ya haya canado el ciclo se cancela.

Inicializando con ciclo comienza a ejecutarse un if donde la condicion es palabrasNoGuardadas. contains(buscar)==true. Si esta instrucción es correcta se le sumara al punteo la cantidad de caracteres que tiene la palabras a los puntos luego hay otra condicion la variable putazo2 tiene que ser igual a cambiar vida si esto se cumple saldra un mensaje “has ganado el juego” y el resultado se lo sumara a una variable y tambien se concatenara el nombre del jugador en una varibale string.

En caso contrario si el jugador se equivoca perdera 5 puntos de los que tiene disponibles al principio se le restara a la variable vida uno cada ves que no acierte una palabra y si vida es igual a cero saldra un mensaje “has perdido el juego ” y se concatenara el nombre del jugador que perdio la partida en una variable striing.

```
while(putaso2<=cambiarVida-1 && vida>0 ){  
  
    String buscar = scanner.next();  
    if( palabrasNoGuardadas.contains(buscar)==true) {  
        putaso2+=1;  
        ++aciertos;  
        puntos+=buscar.length();  
        System.out.println("has encontrado la palabra " + putaso2);  
        if(putaso2==cambiarVida){  
            System.out.println("has ganado la partida");  
            nombresGuardadosGanados+=nombre+"$";  
        };  
    }else{  
        --vida;  
        puntos-=5;  
        System.out.println("te quedan " + vida + " vidas");  
        ++l;  
        if(vida==0){  
            System.out.println("has perdido el juego");  
            nombresGuardados+=nombre+"$";  
        }  
    }  
}
```

Se crea un do while para poder regresar al menú estadístico al ingresar al do se imprimirá los títulos de las opciones y le pedirá que ingrese un numero la variable estadísticas será el que guarde todos los valores que ponga el usuario en consola luego ese resultado se parseara con el método parseInt(). Luego se genera un switch donde la condición es la variable “estadísticasparcear”. Donde el caso 1 representa “historial de partida ” al ingresar a esa opción se ejecutara un for con una condición elemento: palabras todas hace que recorra cada elemento de la matriz.

el caso dos es representable por puntuaciones mal altas las puntuaciones más altas se calcula con el método burbuja donde este método ordena matrices numéricos , en el caso dos se obtiene un if donde la condición neu<=3 se ejecute el nombre de elemento .

en la caso 3 se recorrerá el arreglo por medio de un for y luego imprimirá el elemento con el método System.out.print()

```
do{
    System.out.println("1)historial de partidas 2)puntuaciones mas altas 3)juga
    System.out.println("ingrese el numero de la opcion");
    String estadisticas = scanner.next();
    estadisticasParcear = Integer.parseInt(estadisticas);
    switch (estadisticasParcear){
        case 1:
            String [] palabrastodas = puntosFinal.split("\\$");
            for(String elementostodo: palabrastodas){
                System.out.println(elementostodo);
            }
            break;
        case 2:
            int neu =0;
            String [] palabrase = puntosfinal.split("\\$");
            for(String elementosno: palabrase){
                ++neu;
                if(neu<=3){
                    System.out.println(elementosno);
                }
            }
            break;
        case 3:
            String [] palabrasno = nombresGuardados.split("\\$");
            for(String elementosno: palabrasno){
                System.out.println(elementosno);
            }
            break;
    }
}
```

El caso 4 representa los jugadores que han ganado la perdida al inicializar el caso 4 crea una variable “palabrassi” donde se usara un método de String este método se llama Split() donde nos ayuda separar las concatenaciones que se han creado antes y cada palabra es separada por signo de dólar. Luego se inicia el ciclo for donde recorre el arreglo y irá mostrando cada dato que esta en la nueva matriz.

El caso 5 representa los créditos esta cadena se mostrara en un System.out.println.

Y si es igual a de 6 se cancela todo el do while.

```
        case 4:
            String [] palabrassi = nombresGuardadosGanados.split("\\$");
            for(String elementossi: palabrassi){
                System.out.println(elementossi);
            }
            break;
        case 5:
            System.out.println("KEVIN ALEXIS LOPEZ CONTRERAS 201900157 SECCION D");
            break;
    }

    }while(estadisticasParcear != 6);
}
break;

};
}while(menuParseado!=3);
```