



VectorStock®

VectorStock.com/16489586

Manual de usuario

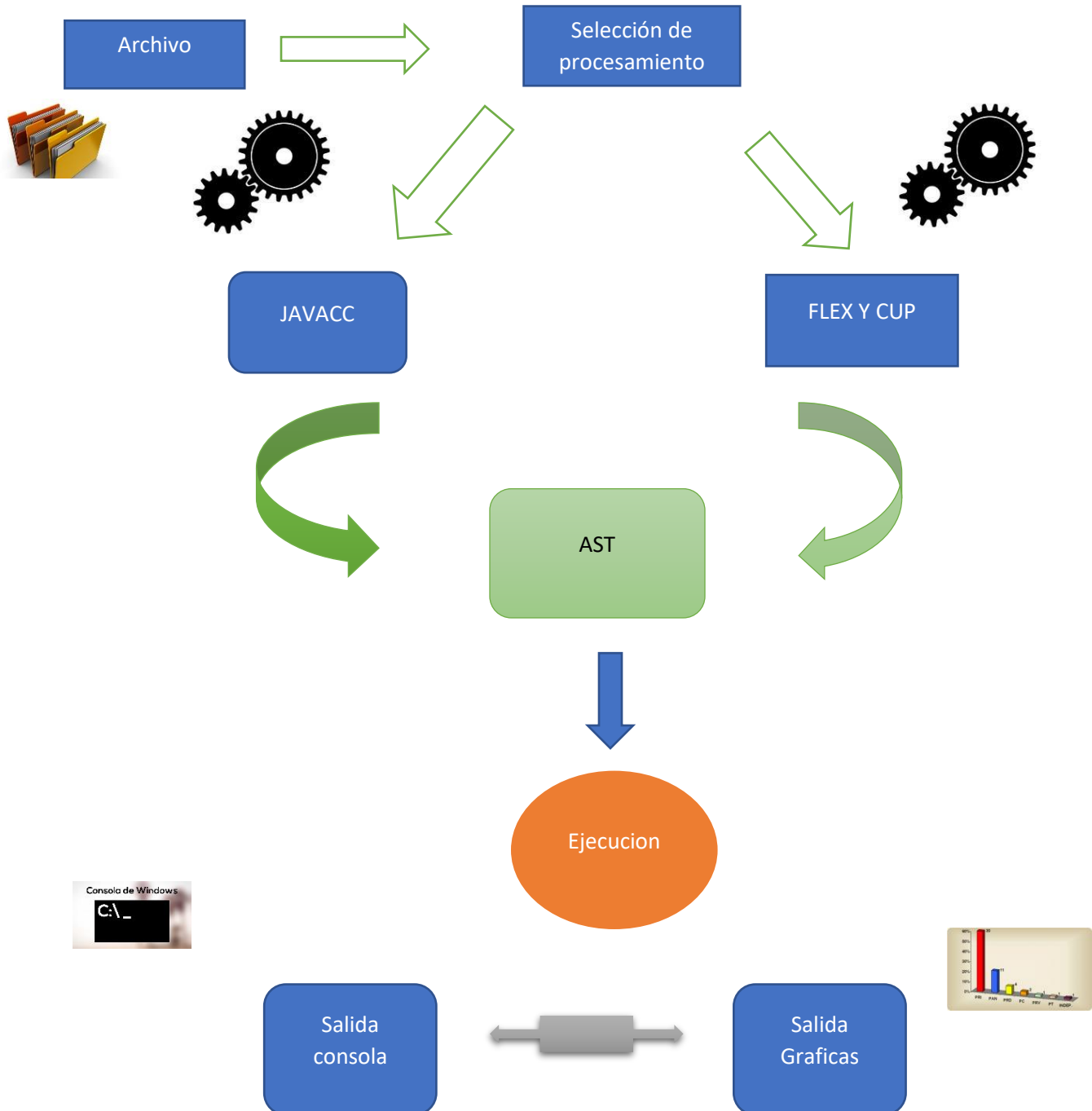
Marvin Ronaldo Martinez Marroquin

201602520

Flujo de la aplicación

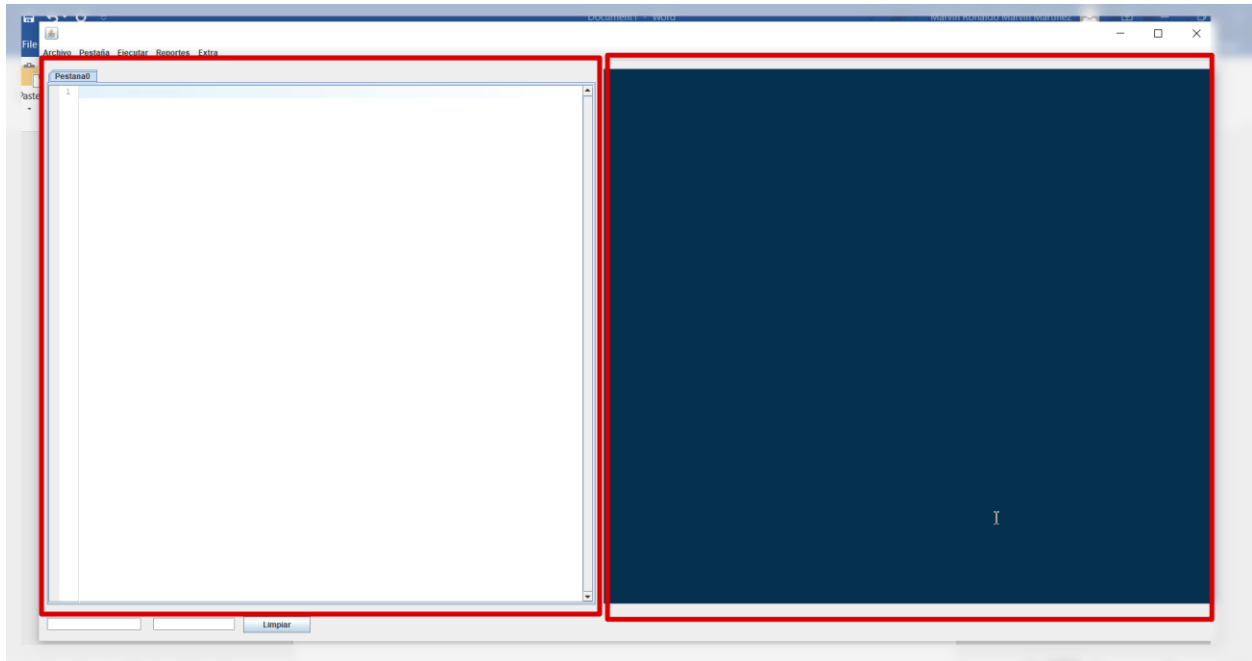
La aplicación puede leer archivos de texto y extensión ARIT. La aplicación inicia con un archivo de entrada el cual se elige por cuál de los 2 medios se desea procesar y luego obtiene los tokens e inicia la forma en construir el árbol.

Luego de que es construido el árbol se inicia a ejecutar las sentencias y el resultado puede ser mostrado en la consola de resultados o visible con graficas si el archivo trae graficas.

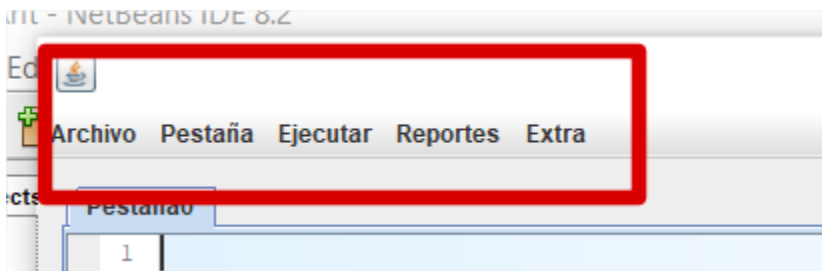


ARIT IDE

Lo primero que nos encontramos en nuestro ide es con un área de texto y una consola de color azul en la cual se mostraran los resultados.

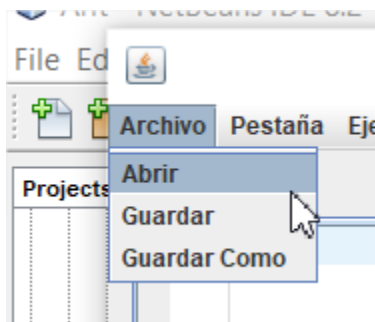


También cuenta con una barra de herramientas.



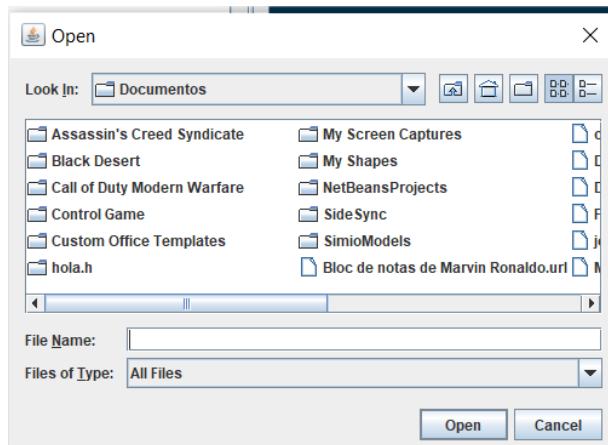
Archivo

En esta función están integradas todas las acciones con respecto a los archivos de arit.

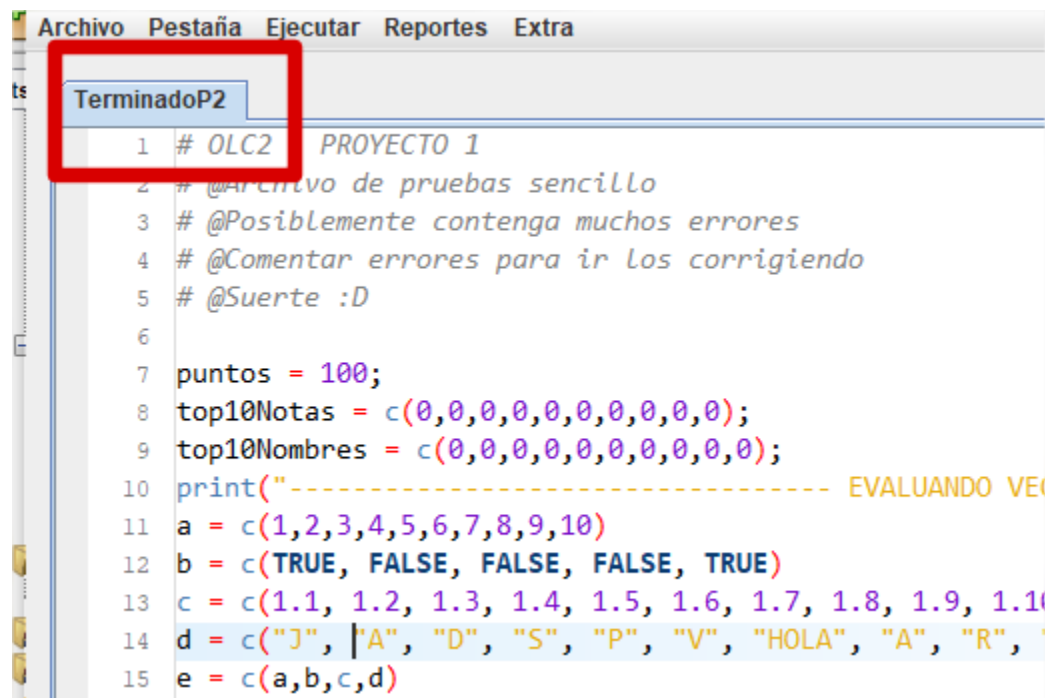


Abrir

Esta función nos permite iniciar con el flujo de nuestro programa ya que permite cargar un archivo de extensión ARIT a nuestra área de texto para ejecutarlo o editarlo.



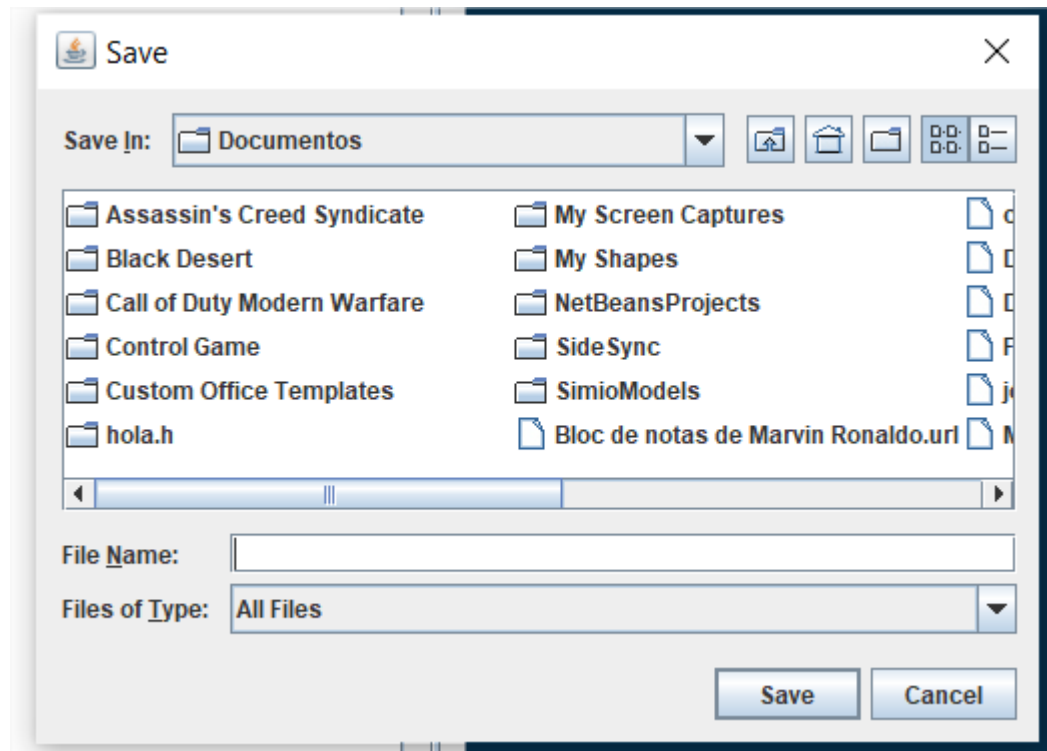
Nos abrirá un cuadro de dialogo en el cual podamos elegir el archivo que queremos cargar. Luego el contenido será plasmado en la pestaña actual.



Automaticamente nos agrega el nombre del archivo para que tengamos en cuenta cual es el que esta abierto en esa pestaña.

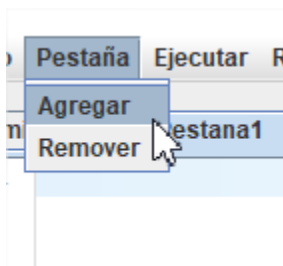
Guardar-Guardar Como

Esta función tiene 2 procesos los cuales depende si la pestaña actual tiene un archivo abierto, si es así automáticamente se guarda el contenido del archivo y si no es así muestra un cuadro de dialogo en donde pregunta la dirección y nombre del archivo para ser guardado.



Pestaña

Esta función cuenta con todo sobre las pestañas del editor.



Agregar

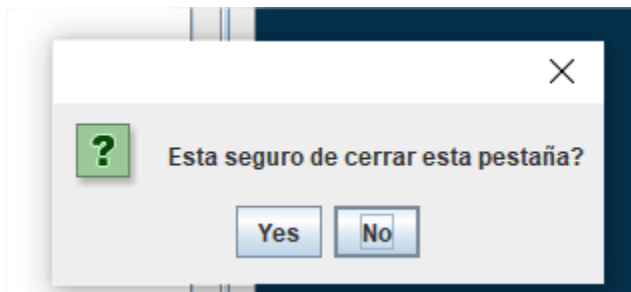
Esta función nos permite agregar una nueva zona de edición de código.



Enumerando con un correlativo para saber cuántas se usaron en esa sesión.

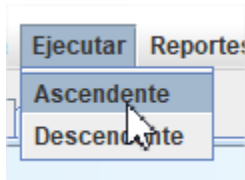
Remover

Esta función quita la pestaña en la que se encuentre preguntando antes si desea guardar los cambios o no.



Ejecutar

En esta función uno selecciona el tipo de interprete que se desea utilizar para el análisis.



Ascendente

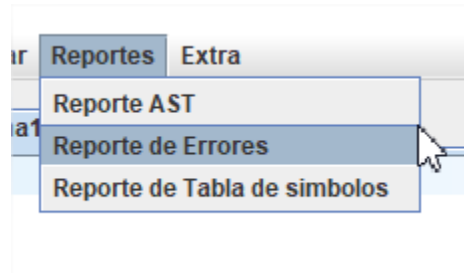
Utiliza JFlex y Cup para el análisis.

Descendente

Utiliza JavaCC para el análisis.

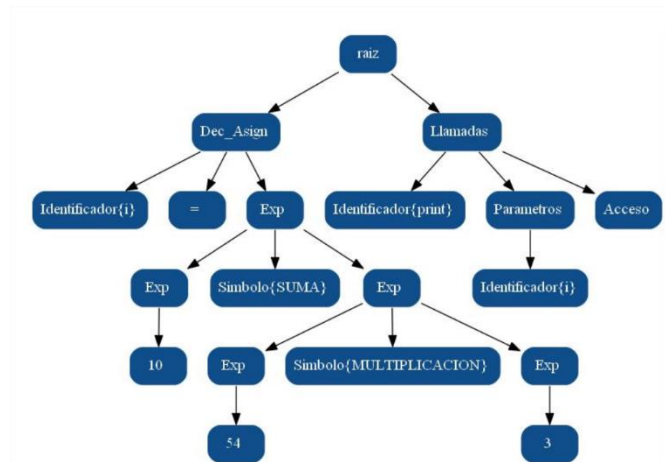
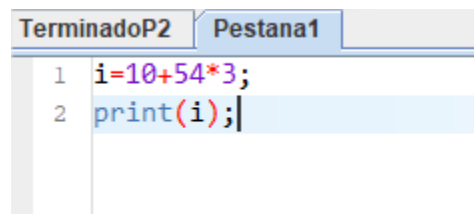
Reportes

En el área de reportes se muestran todos los resultados visuales que no tienen que ver con la consola y graficas.



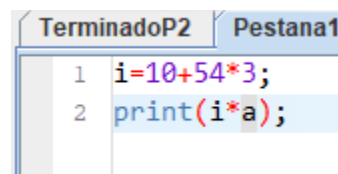
Reporte AST

Este reporte muestra visualmente como se formó el árbol de la entrada que se ejecutó. El reporte esta escrito en graphviz y compilado para una imagen en png.



Reporte de Errores

Este reporte permite ver en una tabla los errores que se detectaron en la ejecución como sus datos mas importantes como línea, tipo de error, columna y mensaje por el cual fue producido.



A screenshot of an error report window. It contains a table with the following data:

Tipo Error	Mensaje	Linea	Columna
SEMANTICO	No se encontro la variable a	2	9

Reporte de Tabla de símbolos

En este reporte cuenta con 2 partes que son una tabla la cual muestra de una manera muy similar que la tabla de errores pero para un mejor comprendimiento se tiene un árbol el cual muestra la jerarquía que tienen las variables.

```
1 i=10+54*3;  
2 if(true){  
3     a=10;  
4     if(true){  
5         b=20;  
6     }  
7 }  
8 print(i);
```

The screenshot displays a Symbol Table Report. On the left, a hierarchical tree structure shows the scope of variables. On the right, a table lists the variables with their types, dimensions, and locations.

Entornos

- Entorno_0
 - Variable_i
 - i
 - vector
 - Location
 - Linea:1
 - Columna:1
 - Dimensiones:1
- Entorno_1
 - Variable_a
 - a
 - vector
 - Location
 - Linea:3
 - Columna:2
 - Dimensiones:1
- Entorno_2
 - Variable_b
 - b
 - vector
 - Location
 - Linea:5
 - Columna:3
 - Dimensiones:1

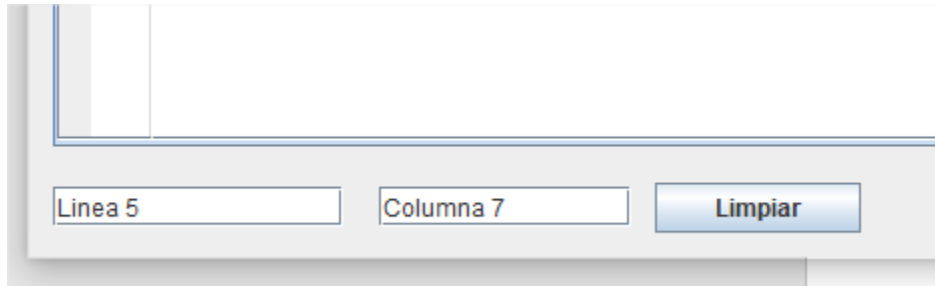
ID	Tipo	Dimensiones	Linea	Columna	Entorno
i	vector	1	1	1	0
a	vector	1	3	2	1
b	vector	1	5	3	2

Extras

Estas funciones permiten visualizar los manuales y diagramas de clases.

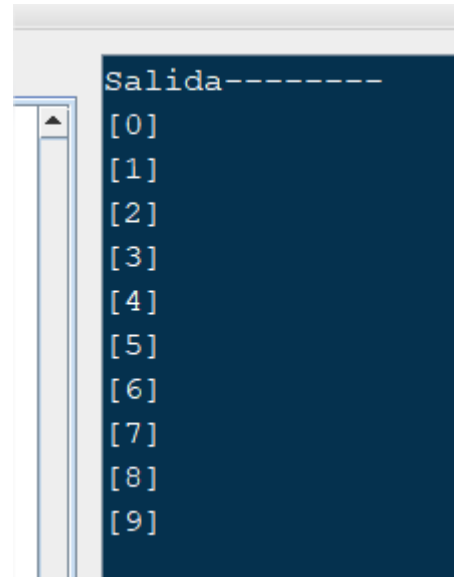
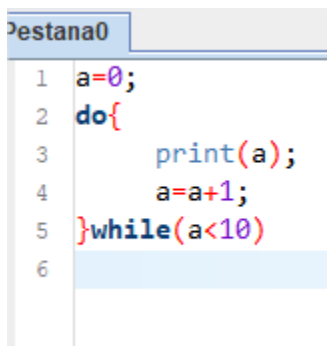
Indicadores

En esta zona del IDE cuenta con dos identificadores que indican en que línea y columna esta situado el cursor. También cuenta con un botón que limpia la consola.



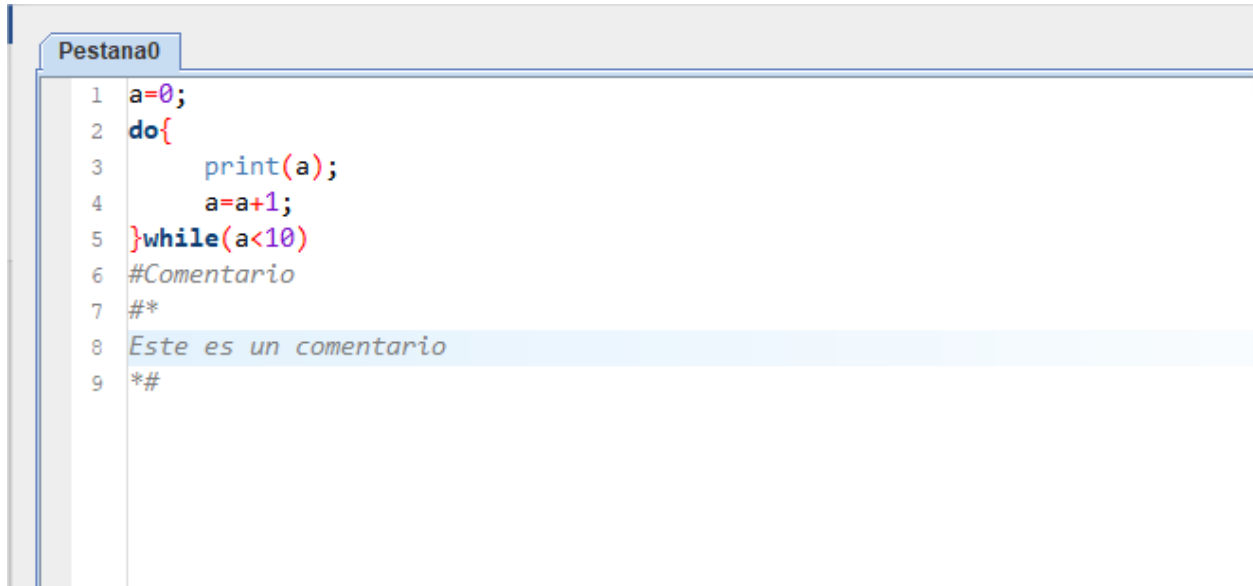
Consola

En ella se muestra toda la salida del programa como errores.



Editor

En esta parte uno cuenta con una barra que enumera las líneas en las que se encuentra y pinta de color las palabras reservadas entre otros.



```
1 a=0;
2 do{
3     print(a);
4     a=a+1;
5 }while(a<10)
6 #Comentario
7 #*
8 Este es un comentario
9 *#
```

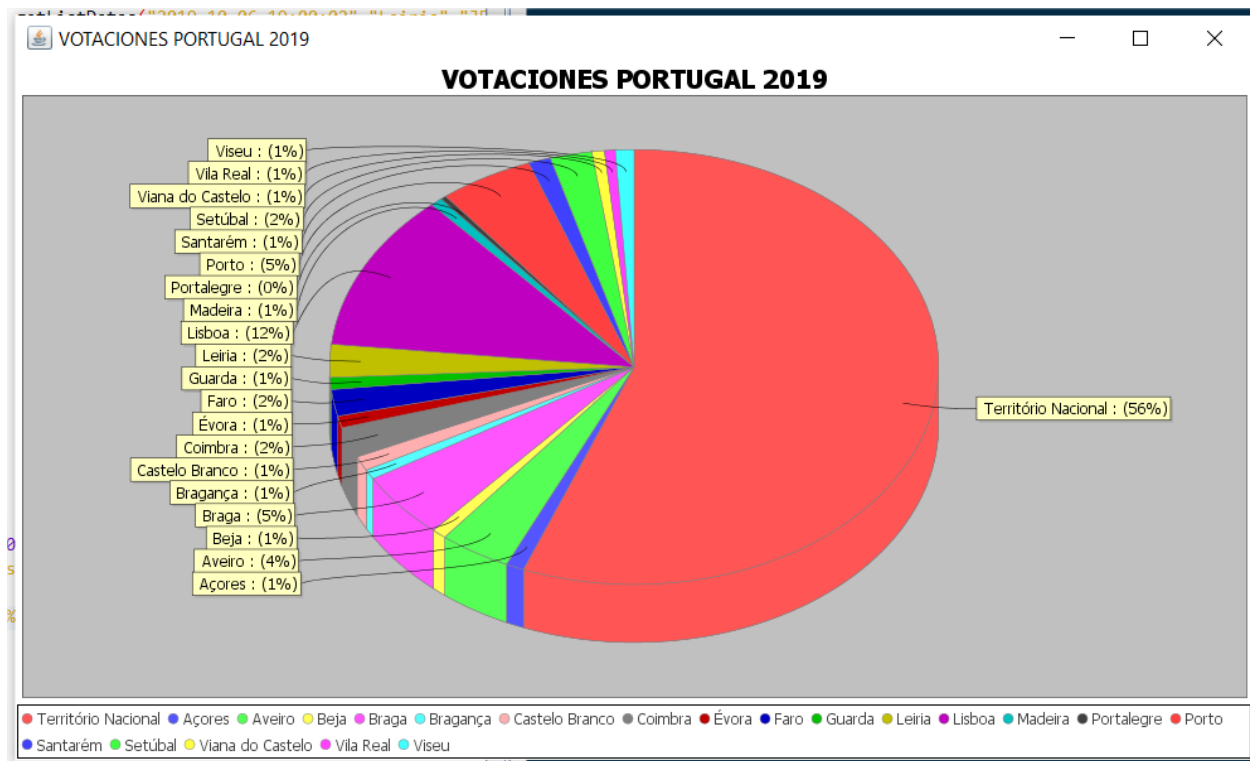
Graficas

El programa cuenta con varios tipos de graficas dependiendo de los datos que se le ingresen.

Pie

Es una gráfica circular que recibe como parámetro un vector de datos, un vector de nombres y un nombre para la gráfica.

```
pie(vectorVotos,vectorNombre,titulo)
```



barPlot

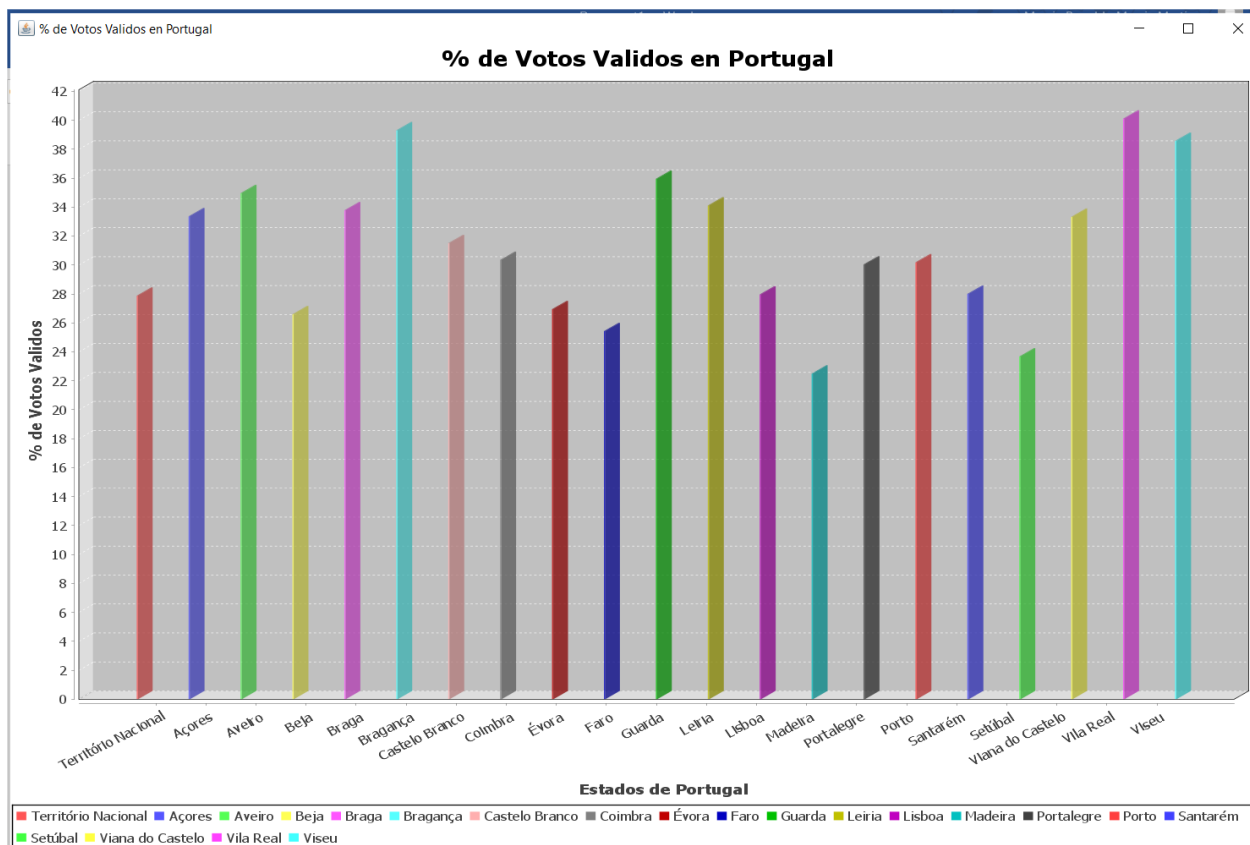
```

barplot(vectorVotos,x,y,titulo,vectorNombre)

porcentajeTerritorio = c(vectorPorcentaje[1], 100)

```

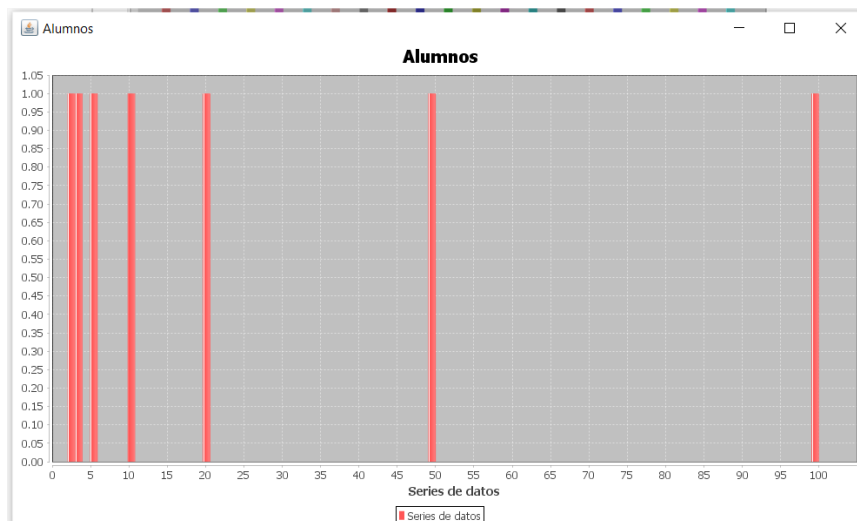
Esta función crea un grafica de barras mandándole como parámetros un vector, nombre del eje x, nombre del eje y, titulo de la grafica y un vector de nombres.



barPlot

En esta grafica recibe como parámetros un vector de datos, un nombre para el eje y el nombre de la grafica.

```
hist(c(10,20,50,5,100,2,3), "Alumnos", "Series de datos");
println("graficando plot sin puntos ");
```



plot

la función plot tiene 3 tipos de uso dependiendo del segundo parámetro, el primero es un vector o matriz, el segundo es un carácter, el tercero es nombre en x, el siguiente es nombre en vertical y el ultimo es el nombre de la grafica

```
println("graficando plot sin lineas...");  
plot(c(10,20,50,5,100,2,3), "i", "Alumnos", "Nota acumulada", "Series de datos");  
println("graficando plot sin puntos...");
```

