# Aplicación de Árboles Binarios de Búsqueda

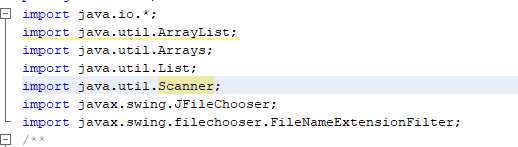
Primero comenzamos explicando de que trata el programa:

El programa tiene como función importar un archivo csv válido el cual contendrá los datos importantes de una factura y los guardará en un árbol binario, para el cual se podrá consultar por una identificación única durante la ejecución del programa

Proceso de funcionamiento

# Importación de librerías

Primero se importan todas las librerías necesarias para el funcionamiento del programa



# Búsqueda de archivo csv

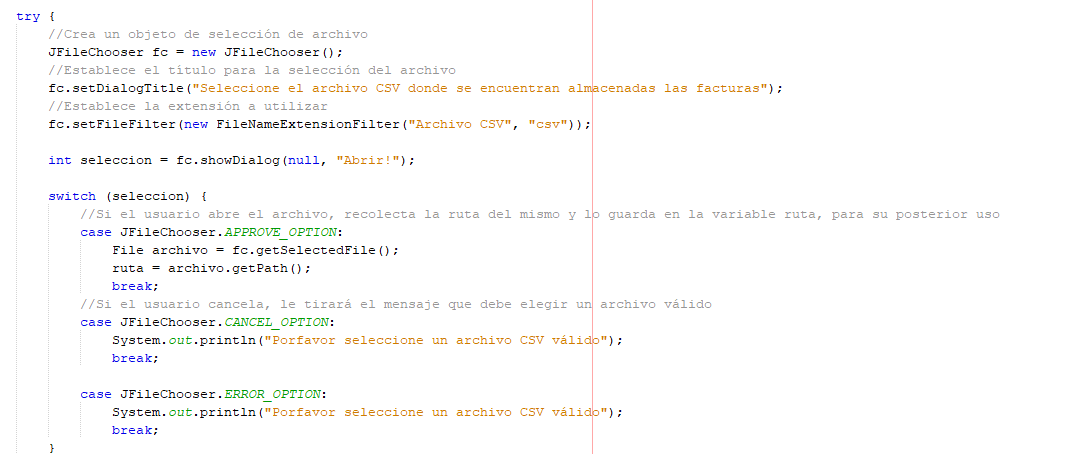
## Inicializando variables

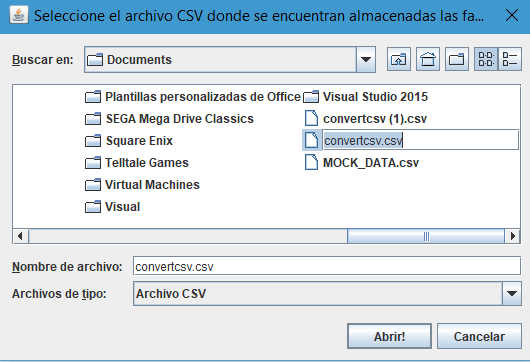
Segundo al iniciar la ejecución del programa es necesario saber la ubicación del archivo principal csv, primero se inicializa una variable de la ruta del archivo



## 2.2 Llamado a la ventana de selección

De allí se crea toda la parte necesaria el cual llamará a una ventana de abrir archivo preguntándole al usuario dónde está el archivo, y los 3 posibles escenarios del mismo.





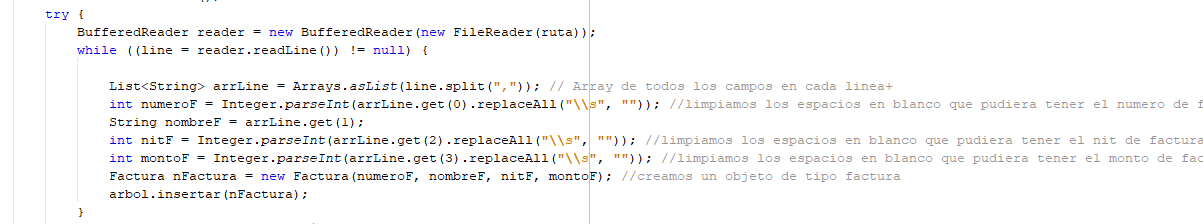
# Inicialización de objetos

A partir de allí se inicializa el objeto y variable necesario para guardar los elementos en árboles.



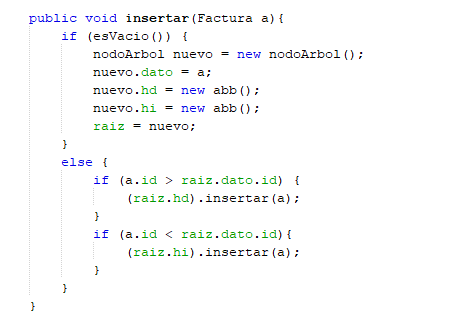
# Guardado de objetos en el árbol

Crea un BufferedReader para leer comandos del usuario y crea un objeto de tipo Factura, y también se limpia todos los espacios en blanco, todo se realiza por cada línea del csv y luego se insertan al árbol, el proceso termina cuando ya no hay más datos que añadir.



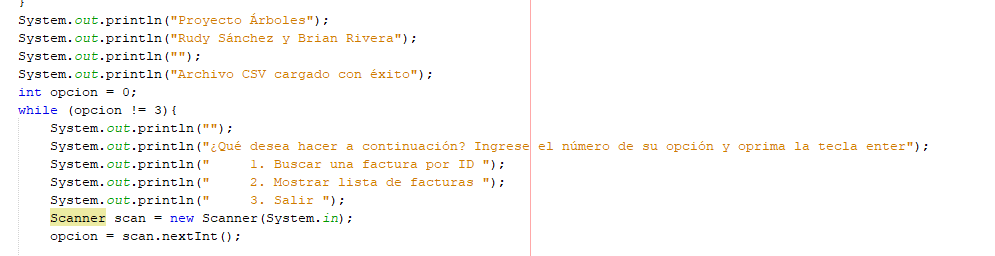
Al utilizar la función arbol.insertar() se emplea una función recursiva el cual revisa en qué lado del nodo se va a insertar y luego termina insertando el objeto.

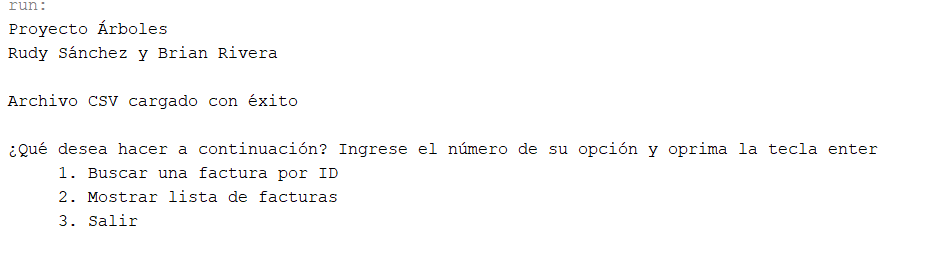
## 4.1 Funcionamiento del método insertar



# Bienvenida al usuario

Una vez realizado todo el trabajo de fondo y si todo sale de manera satisfactoria se le presenta al usuario buscar una factura o bien buscar todas las facturas existentes.

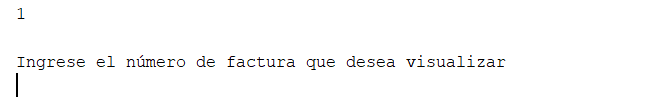


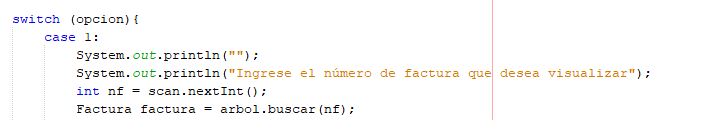


# Posibles escenarios

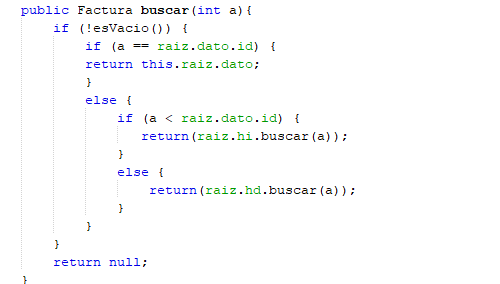
## 6.1 Escenario 1 (Búsqueda específica por ID)

Si el usuario elige la primera opción se le preguntará por la ID de factura y luego efectuará la búsqueda

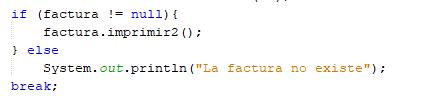


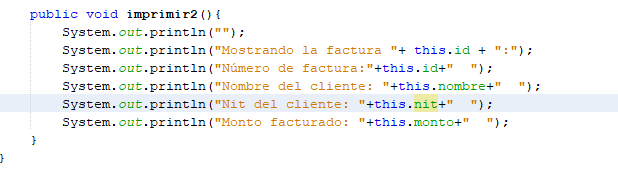


La función árbol.buscar() también es recursivo, buscará por cada uno de los nodos hasta encontrar si existe una correcta

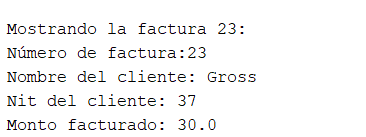


Si la factura existe se le llama a la función de imprimir los datos correspondientes para el usuario, si la factura no existe entonces simplemente dirá al usuario que la factura no existe.

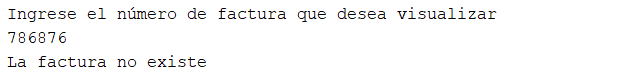




**Si la factura existe:**

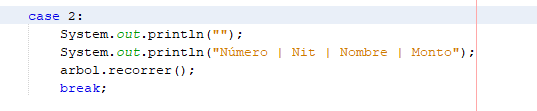


**Si la factura no existe:**

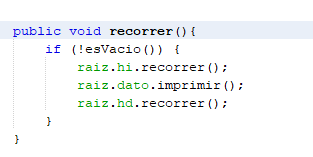


## 6.2 Escenario 2 (Imprimir todo)

Primero se imprime una línea de texto y llama a la función recorrer del árbol



La función de recorrido es recursiva, y también utiliza la función de imprimir por cada factura que exista.





## 6.3 Escenario 3 (Salir)

Detiene la ejecución del programa



## 6.4 Escenario 4 (Dato incorrecto)



Se le indicará al usuario que debe seleccionar una opción correcta

