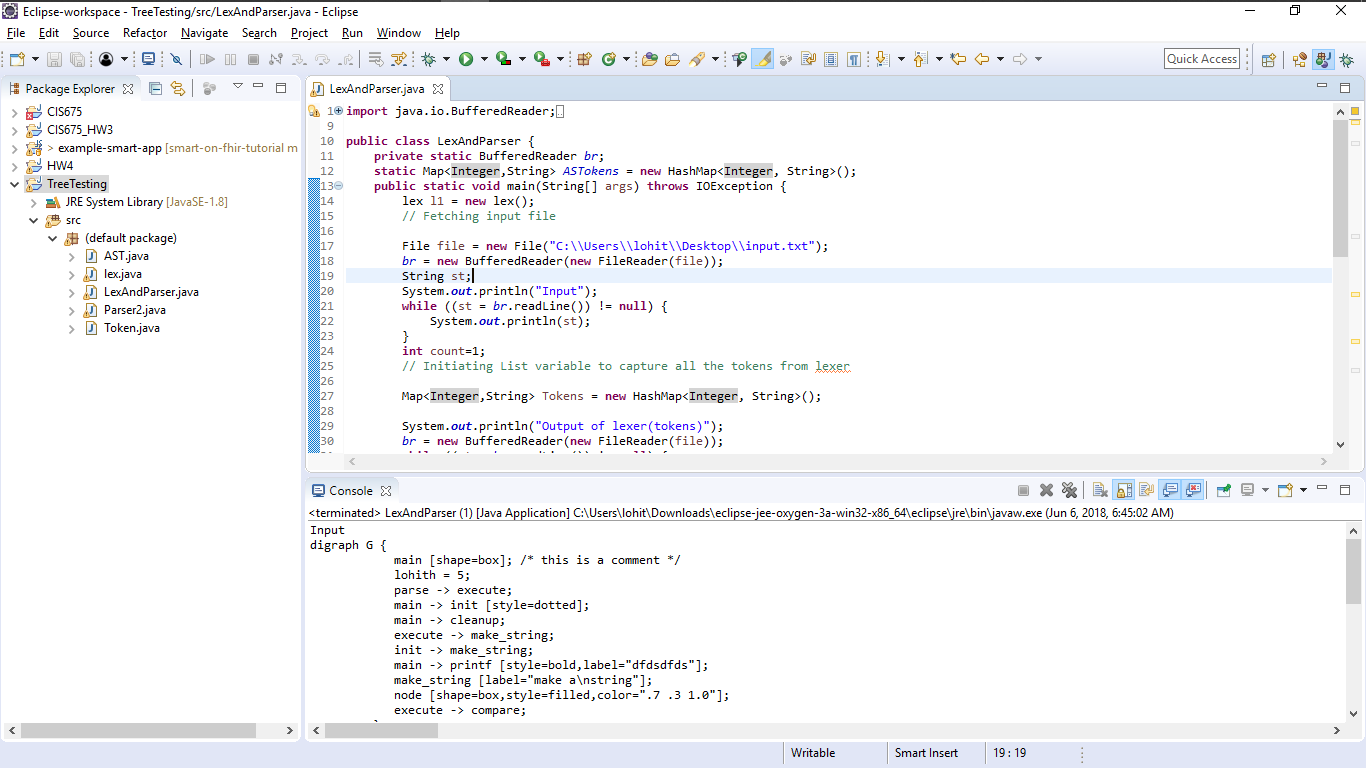
CIS675-Hw4 Submitted by : Lohith Nimmala

Code at : https://github.com/Lohith499/CIS675\_HW4\_Parser\_with\_nodes 

**Output**

Input

digraph G {

main [shape=box]; /\* this is a comment \*/

lohith = 5;

parse -> execute;

main -> init [style=dotted];

main -> cleanup;

execute -> make\_string;

init -> make\_string;

main -> printf [style=bold,label="dfdsdfds"];

make\_string [label="make a\nstring"];

node [shape=box,style=filled,color=".7 .3 1.0"];

execute -> compare;

}

Output of lexer(tokens)

1.DIGRAPH,digraph 2.ID,G 3.LBRACE,{

4.ID,main 5.LBRACKET,[ 6.ID,shape 7.EQUAL,= 8.ID,box 9.RBRACKET,] 10.SEMICOLON,;

11.ID,lohith 12.EQUAL,= 13.ID,5 14.SEMICOLON,;

15.ID,parse 16.EDGEOP,-> 17.ID,execute 18.SEMICOLON,;

19.ID,main 20.EDGEOP,-> 21.ID,init 22.LBRACKET,[ 23.ID,style 24.EQUAL,= 25.ID,dotted 26.RBRACKET,] 27.SEMICOLON,;

28.ID,main 29.EDGEOP,-> 30.ID,cleanup 31.SEMICOLON,;

32.ID,execute 33.EDGEOP,-> 34.ID,make\_string 35.SEMICOLON,;

36.ID,init 37.EDGEOP,-> 38.ID,make\_string 39.SEMICOLON,;

40.ID,main 41.EDGEOP,-> 42.ID,printf 43.LBRACKET,[ 44.ID,style 45.EQUAL,= 46.ID,bold 47.COMMA, 48.ID,label 49.EQUAL,= 50.ID,ID 51.RBRACKET,] 52.SEMICOLON,;

53.ID,make\_string 54.LBRACKET,[ 55.ID,label 56.EQUAL,= 57.ID,ID 58.RBRACKET,] 59.SEMICOLON,;

60.NODE,node 61.LBRACKET,[ 62.ID,shape 63.EQUAL,= 64.ID,box 65.COMMA, 66.ID,style 67.EQUAL,= 68.ID,filled 69.COMMA, 70.ID,color 71.EQUAL,= 72.ID,ID 73.RBRACKET,] 74.SEMICOLON,;

75.ID,execute 76.EDGEOP,-> 77.ID,compare 78.SEMICOLON,;

79.RBRACE,}

Abstract Syntax Tree

(GRAPH (DIGRAPH G LBRACE (Stmt\_list ((stmt ((Node\_Stmt ((Node\_id (main)) (Attr\_List (LBRACKET (A\_list (shape EQUAL box)) RBRACKET)))))) SEMICOLON (stmt (lohith EQUAL 5)) SEMICOLON (stmt (SEMICOLON (EDGE\_Stmt ((Node\_id (parse)) (Edge\_RHS (EDGEOP (Node\_id (execute)))))))) SEMICOLON (stmt (SEMICOLON (EDGE\_Stmt ((Node\_id (main)) (Edge\_RHS (EDGEOP (Node\_id (init)))) (Attr\_list (LBRACKET (A\_list (style EQUAL dotted)) RBRACKET)))))) SEMICOLON (stmt (SEMICOLON (EDGE\_Stmt ((Node\_id (main)) (Edge\_RHS (EDGEOP (Node\_id (cleanup)))))))) SEMICOLON (stmt (SEMICOLON (EDGE\_Stmt ((Node\_id (execute)) (Edge\_RHS (EDGEOP (Node\_id (make\_string)))))))) SEMICOLON (stmt (SEMICOLON (EDGE\_Stmt ((Node\_id (init)) (Edge\_RHS (EDGEOP (Node\_id (make\_string)))))))) SEMICOLON (stmt (SEMICOLON (EDGE\_Stmt ((Node\_id (main)) (Edge\_RHS (EDGEOP (Node\_id (printf)))) (Attr\_list (LBRACKET (A\_list (style EQUAL bold COMMA label EQUAL ID)) RBRACKET)))))) SEMICOLON (stmt ((Node\_Stmt ((Node\_id (make\_string)) (Attr\_List (LBRACKET (A\_list (label EQUAL ID)) RBRACKET)))))) SEMICOLON (stmt (SEMICOLON (Attr\_Stmt (NODE (Attr\_list (LBRACKET (A\_list (shape EQUAL box COMMA style EQUAL filled COMMA color EQUAL ID)) RBRACKET)))))) SEMICOLON (stmt (SEMICOLON (EDGE\_Stmt ((Node\_id (execute)) (Edge\_RHS (EDGEOP (Node\_id (compare)))))))) SEMICOLON)) RBRACE))

Status of Parser execution = Program has no Syntax Errors