EX.NO:2 IMPLEMENTATION OF LEXICAL ANALYZER USING LEX TOOL

Aim:

To implement lexical analyser using lex tool

CODE:

```
%{
  #include<stdio.h>
%}

%option noyywrap

delim   [ \t\n]
ws              {delim}+
letter    [a-zA-Z]
digit     [0-9]
identifier  {letter}({letter}|{digit}|_)*
number          {digit}+(\.{digit}+)?(E[+|-]?{digit}+)?
sept      [,.:;!?(){}]
nonblank [^ \t]
string    \".*\"

%%

#.* {printf("\n%s    -    Preprocessor directive",yytext);}

int|float|char|double|while|for|struct|typedef|do|if|break|continue|void|switch|ret
urn|else|goto    {printf("\n%s    -    Keyword",yytext);}

{identifier}\(({string}|{identifier}|,)*\)   {printf("\n%s    -    Function
call",yytext);}

{sept}{printf("\n%s    -    Separator",yytext);}

{identifier} {printf("\n%s    -    Identifier",yytext);}
```

```
{number} {printf("\n%s    -    Number",yytext);}

\= {printf("\n%s     -    Assignment Operator",yytext);}

\+ |
\- |
\* |
\/   { printf("\n%s    -    Arithmetic Operator",yytext);}

\<= |
\>= |
\< |
\> |
== {printf("\n%s    -    Relational Operator",yytext);}

%%

int main()
{
        FILE * fd = fopen("/home/csec86/cdlab/ex2/s1.txt","r");
        if(!fd)
                printf("\nError opening file");
        yyin=fd;
        yylex();
        printf("\n");
}
```

INPUT FILE:

```
main()
{
int var=10,b=20,c;
float x=10.0,y=20,z;
if(a>b)
        printf("a is greater");
else
        printf("b is greater");
```

}

OUTPUT:

```
csec86@ccl-06:~/cdlab/ex2$ lex lex_ex.l
csec86@ccl-06:~/cdlab/ex2$ gcc lex.yy.c
csec86@ccl-06:~/cdlab/ex2$ ./a.out

main()    -    Function call

{    -    Separator

int    -    Keyword
var    -    Identifier
=     -    Assignment Operator
10    -    Number
,    -    Separator
b    -    Identifier
=     -    Assignment Operator
20    -    Number
,    -    Separator
c    -    Identifier
;    -    Separator

float    -    Keyword
x    -    Identifier
=     -    Assignment Operator
10.0    -    Number
,    -    Separator
y    -    Identifier
=     -    Assignment Operator
20    -    Number
,    -    Separator
z    -    Identifier
;    -    Separator

if    -    Keyword
(    -    Separator
a    -    Identifier
>    -    Relational Operator
b    -    Identifier
)    -    Separator
```

```
float    -    Keyword
x    -    Identifier
=     -    Assignment Operator
10.0    -    Number
,    -    Separator
y    -    Identifier
=     -    Assignment Operator
20    -    Number
,    -    Separator
z    -    Identifier
;    -    Separator

if    -    Keyword
(    -    Separator
a    -    Identifier
>    -    Relational Operator
b    -    Identifier
)    -    Separator

printf("a is greater")    -    Function call
;    -    Separator

else    -    Keyword

printf("b is greater")    -    Function call
;    -    Separator

}    -    Separator

csec86@ccl-06:~/cdlab/ex2$
```