```
single_input: {
            return(1);
        }

small_stmt: expire_stmt
            {
                $$ = $1;
            }
        | PRINT
            {
                $$ = new PrintStatement;
            }
        | PRINT
          test
            {
                $$ = new PrintStatement($2);
            }
        | test
          EQUALS
          test
            {
                $$ = newExpressionStatement(new AssignmentExpression($1, $3));
            }

expr_stmt     : test
            {
                $$ = new Expression


or_test     : and_test
            {
                $$=$1;
            }
        | or_test
          OR
          and_test
            {
                $$ = new BinaryExpression(OR_OP, $1, $3);
            }



OR_OP
AND_OP
NOT_OP
LESSER_OP
```

suite: simple_stmt | NEWLINE INDENT stmt+ DEDENT

how do you do "stmt+"?

stmt_list: stmt
         | stmt_list   stmt

set stmt_list = to stmt+

do not have to do the following operators:
IGNORE PERCENT
IGNORE SLAHS SLAHS OP
IGNORE STAR STAR

all a test can be is an or_test, you can ignore the ['if' or_test 'else' test] | lamedf]

still have to figure out how to make a simple statement from a small statement
^in the simple_stmt python grammar

still have to figure out test: