

Test Cases - Software Testing and Maintenance

open_character_stream:

Input	Test Path Traversal	Expected Output
Fname = null	[1,2,5]	null
Fname = filename (any filename)	[1, 3, 4, 5]	filename

get_char:

Input	Test Path Traversal	Expected Output
Empty or non empty file	[1,2,3,4]	Contents of file

unget_char:

Input	Test Path Traversal	Expected Output
br can be null or any other value.	[1,2,]	br = null

open_token_stream:

Input	Test Path Traversal	Expected Output
Fname = null or fname = “”	[1, 2, 4]	fname = null
Fname = “Any kind of file name”	[1, 3, 4]	fname = filename

get_token:

Input	Test Path Traversal	Expected Output
Nothing from first get_char	[1, 2, 3]	null
Nothing from second get_char	[1, 2, 4, 5, 6, 7]	null
Special symbol found	[1, 2, 3, 4, 5, 6, 8, 9]	Any special symbol found in sb
Nothing from third get_char	[1, 2, 4, 5, 6, 8, 10, 12, 14, 15]	Contents of sb

(Neither/String/Comment)	[1, 2, 4, 5, 6, 8, 10, 11, 14, 15] [1, 2, 4, 5, 6, 8, 10, 12, 13, 14, 15]	
Nothing from fourth get_char (Neither/String/Comment, Could also be end of file)	[1, 2, 4, 5, 6, 8, 10, 12, 14, 16, 17, 19, 20, 21] [1, 2, 4, 5, 6, 8, 10, 11, 14, 16, 17, 19, 20, 21] [1, 2, 4, 5, 6, 8, 10, 12, 13, 14, 16, 17, 19, 20, 21] [1, 2, 4, 5, 6, 8, 10, 12, 14, 16, 17, 18, 20, 21] [1, 2, 4, 5, 6, 8, 10, 11, 14, 16, 17, 18, 20, 21] [1, 2, 4, 5, 6, 8, 10, 12, 13, 14, 16, 17, 18, 20, 21]	Contents of sb up until now
Another special symbol found	[1, 2, 4, 5, 6, 8, 10, 12, 14, 16, 17, 19, 20, 22, 23] [1, 2, 4, 5, 6, 8, 10, 11, 14, 16, 17, 19, 20, 22, 23] [1, 2, 4, 5, 6, 8, 10, 12, 13, 14, 16, 17, 19, 20, 22, 23] [1, 2, 4, 5, 6, 8, 10, 12, 14, 16, 17, 18, 20, 22, 23] [1, 2, 4, 5, 6, 8, 10, 11, 14, 16, 17, 18, 20, 22, 23] [1, 2, 4, 5, 6, 8, 10, 12, 13, 14, 16, 17, 18, 20, 22, 23]	Sb contents so far
Some string	[1, 2, 4, 5, 6, 8, 10, 12, 14, 16, 17, 19, 20, 22, 24, 25, 26, 27] [1, 2, 4, 5, 6, 8, 10, 11, 14, 16, 17, 19, 20, 22, 24, 25, 26, 27] [1, 2, 4, 5, 6, 8, 10, 12, 13, 14, 16, 17, 19, 20, 22, 24, 25, 26, 27] [1, 2, 4, 5, 6, 8, 10, 12, 14, 16, 17, 18, 20, 22, 24, 25, 26, 27] [1, 2, 4, 5, 6, 8, 10, 11, 14, 16, 17, 18, 20, 22, 24, 25, 26, 27] [1, 2, 4, 5, 6, 8, 10, 12, 13, 14, 16, 17, 18, 20, 22, 24, 25, 26, 27]	Sb content
Neither string nor comment	[1, 2, 4, 5, 6, 8, 10, 12, 14, 16, 17, 19, 20, 22, 24, 28, 29] [1, 2, 4, 5, 6, 8, 10, 11, 14, 16, 17, 19, 20, 22, 24, 28, 29]	Sb content

	[1, 2, 4, 5, 6, 8, 10, 12, 13, 14, 16, 17, 19, 20, 22, 24, 28, 29] [1, 2, 4, 5, 6, 8, 10, 12, 14, 16, 17, 18, 20, 22, 24, 28, 29] [1, 2, 4, 5, 6, 8, 10, 11, 14, 16, 17, 18, 20, 22, 24, 28, 29] [1, 2, 4, 5, 6, 8, 10, 12, 13, 14, 16, 17, 18, 20, 22, 24, 28, 29]	
Normal token	[1, 2, 4, 5, 6, 8, 10, 12, 14, 16, 17, 19, 20, 22, 24, 28, 30] [1, 2, 4, 5, 6, 8, 10, 11, 14, 16, 17, 19, 20, 22, 24, 28, 30] [1, 2, 4, 5, 6, 8, 10, 12, 13, 14, 16, 17, 19, 20, 22, 24, 28, 30] [1, 2, 4, 5, 6, 8, 10, 12, 14, 16, 17, 18, 20, 22, 24, 28, 30] [1, 2, 4, 5, 6, 8, 10, 11, 14, 16, 17, 18, 20, 22, 24, 28, 30] [1, 2, 4, 5, 6, 8, 10, 12, 13, 14, 16, 17, 18, 19, 20, 22, 24, 28, 30]	Sb content

is_token_end:

Input	Test Path Traversal	Expected Output
Null value	[1,2]	true
Any string ending with a "", new line, return, or tab	[1,3,4,5]	true
Any string that doesn't end with those characters such as "this_is_a_string"	[1,3,4,6]	false
//comment\n, //comment\t, //comment\r	[1,3,7,8,9]	true
//comment	[1,3,7,8,10]	false
(,),[], `,"",`	[1,3,7,11,12]	true
+, -, *, 5, a, ... (not a special symbol) with a \t,\r or \n at the end	[1,3,7,11,13, 14]	true

+, -, *, 5, a, ... (not a special symbol) with more code after	[1,3,7,11,13]	false
--	---------------	-------

token_type:

Input	Test Path Traversal	Expected Output
Just keywords	[1, 2]	keyword
Just special symbol	[1, 3, 4]	Special symbol
Just identifier	[1, 3, 5, 6]	identifier
Just constant number	[1, 3, 5, 7, 8]	Constant number
Just constant string	[1, 3, 5, 7, 9, 10]	Constant string
Just constant char	[1, 3, 5, 7, 9, 11, 12]	Constant char
Just comment	[1, 3, 5, 7, 9, 11, 13, 14]	comment
None of the above	[1, 3, 5, 7, 9, 11, 13]	returns nothing

print_token:

Input	Test Path Traversal	Expected Output
Error token (None of the below)	[1, 2]	ex: error, 32a
Keyword token	[1, 3, 4]	ex: keyword, and
Special symbol	[1, 3, 5, 6]	ex: (
Identifier token	[1, 3, 5, 7, 8]	ex: identifier, a1
Constant number token	[1, 3, 5, 7, 9, 10]	ex: numeric, 12
Constant string token	[1, 3, 5, 7, 9, 11, 12]	ex: string, asd
Constant character token	[1, 3, 5, 7, 9, 11, 13, 14]	ex: character,\#a\
Comment token	[1, 3, 5, 7, 9, 11, 13, 15, 16]	ex: comment,\;123\

is_comment:

Input	Test Path Traversal	Expected Output
Set ch = 59 at index 0	[1,2]	true
ch != 59 at index 0	[1,3]	false

is_keyword:

Input	Test Path Traversal	Expected Output
or, and, xor, =>, if, lambda	[1,2]	true
Any other character	[1,3]	false

is_char_constant:

Input	Test Path Traversal	Expected Output
#a,#b,#c, etc.	[1,2]	true
@a, 1b, -c, #3,... (any character not # in the first index or no letter after #)	[1,3]	false

is_num_constant:

Input	Test Path Traversal	Expected Output
a,v,@,%,< (any non-numeric value)	[1,7]	false
12#45, 1\$2, 2fr\$3 (any non-integer characters between the numbers or at the end)	[1,2,3,5]	false
1234,123,1255, etc.	[1,2,3,4,2,6]	true
1,2,5,... (any single number constants)	[1,2,6]	true

is_str_constant:

Input	Test Path Traversal	Expected Output
123, string (not starting with "")	[1,7]	false
""	[1,2,3,4], [1,2,6]	true
"123", "abc"	[1,2,3,5,2,6]	true

is_identifier:

Input	Test Path Traversal	Expected Output
1a, #s, 4c,.... (doesn't begin with a letter)	[1,2,6]	false
a, aa, a23,....	[1,2,3,4,2,6], [1,7]	true
a#,a12\$3a,.... (a symbol in between)	[1,2,3,5]	false

print_spec_symbol:

Input	Test Path Traversal	Expected Output
)],[;`,""	[1,3,5,7,9,11,13], [1,2],[1,3,4],[1,3,5,6],[1,3,5,7,8],[1 ,3,5,7,9,10],[1,3,5,7,9,11,12],[1,3,5 ,7,9,11,13,14]	Any string that describes the given input

is_spec_symbol:

Input	Test Path Traversal	Expected Output
)],[;`,""	[1,3,5,7,9,11,13,15], [1,2],[1,3,4],[1,3,5,6],[1,3,5,7,8],[1 ,3,5,7,9,10],[1,3,5,7,9,11,12],[1,3,5 ,7,9,11,13,14]	True
Anything else	_,-,w,3,....	False

Main function

Input	Test Path Traversal	Expected Output
java Printtokens	[1,2,6,7]	fname = “”
java Printtokens test	[1,3,4,6,7]	fname = test
Any other kind of input	[1,3,5]	Error message