Started on	Monday, 12 June 2023, 1:44 PM
State	Finished
Completed on	Monday, 12 June 2023, 2:27 PM
Time taken	42 mins 37 secs
Marks	2.70/3.00
Grade	9.00 out of 10.00 (90 %)

Question 1

Correc

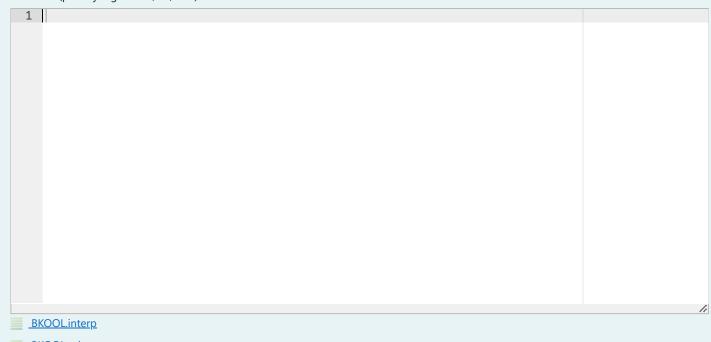
Mark 1.00 out of 1.00

Use ANTLR to write regular expressions describing a Pascal identifier that must begin with a lowercase letter ('a' to 'z'), but may continue with many characters which are lowercase letter or digit ('0' to '9').

For example:

Test	Result
"abc"	abc, <eof></eof>

Answer: (penalty regime: 10, 20, ... %)



BKOOL.tokens

BKOOLLexer.interp

BKOOLLexer.py

BKOOLLexer.tokens

BKOOLParser.pyBKOOLVisitor.py

	Test	Expected	Got	
~	"abc"	abc, <eof></eof>	abc, <eof></eof>	~
~	"abc a12"	abc,a12, <e0f></e0f>	abc,a12, <e0f></e0f>	~
~	"abc A12"	abc,Error Token A	abc,Error Token A	~
~	"abc?svn"	abc,Error Token ?	abc,Error Token ?	~
~	"0a12"	Error Token 0	Error Token 0	~

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question 2

Correc

Mark 0.80 out of 1.00

Use ANTLR to write regular expressions describing Pascal tokens For a number to be taken as "real" (or "floating point") format, it must either have a decimal point, or use scientific notation. For example, 1.0, 1e-12, 1.0e-12, 0.000000001 are all valid reals. At least one digit must exist on either side of a decimal point.

For example:

Test	Result
"1.0"	1.0, <eof></eof>

Answer: (penalty regime: 10, 20, ... %)

1 ||

BKOOL.interp

BKOOL.tokens

BKOOLLexer.interp

BKOOLLexer.py

BKOOLLexer.tokens

BKOOLParser.py

BKOOLVisitor.py

	Test	Expected	Got	
~	"1.0"	1.0, <eof></eof>	1.0, <eof></eof>	~
~	"1e-12"	1e-12, <e0f></e0f>	1e-12, <e0f></e0f>	~
~	"1.0e-12"	1.0e-12, <eof></eof>	1.0e-12, <eof></eof>	~
~	"0.001"	0.001, <eof></eof>	0.001, <eof></eof>	~
~	".01"	Error Token .	Error Token .	~
~	"'0.01'"	Error Token '	Error Token '	~

	Test	Expected	Got	
~	"1.0e"	1.0,Error Token e	1.0,Error Token e	~
~	"0e123"	0e123, <e0f></e0f>	0e123, <e0f></e0f>	~
~	".e123"	Error Token .	Error Token .	~
~	"e123"	Error Token e	Error Token e	~

Passed all tests! 🗸



Marks for this submission: 1.00/1.00. Accounting for previous tries, this gives **0.80/1.00**.

Question 3

Correc

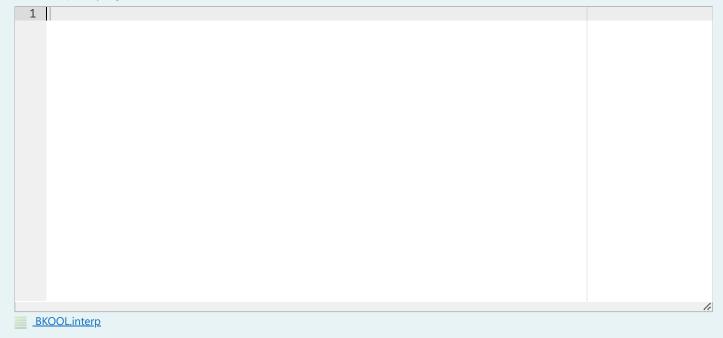
Mark 0.90 out of 1.00

Use ANTLR to write regular expressions describing Pascal strings are made up of a sequence of characters between single quotes: 'string'. The single quote itself can appear as two single quotes back to back in a string: 'isn''t'.

For example:

Test	Result
"'Yanxi Palace - 2018'"	'Yanxi Palace - 2018', <eof></eof>

Answer: (penalty regime: 10, 20, ... %)



- BKOOL.tokens
- BKOOLLexer.interp
- BKOOLLexer.py
- BKOOLLexer.tokens
- BKOOLParser.py
- BKOOLVisitor.py

		Test	Expected	Got	
	~	"'Yanxi Palace - 2018'"	'Yanxi Palace - 2018', <eof></eof>	'Yanxi Palace - 2018', <eof></eof>	~
	~	"'abc?svn'"	'abc?svn', <e0f></e0f>	'abc?svn', <e0f></e0f>	~
	~	"'1.0e-12'"	'1.0e-12', <eof></eof>	'1.0e-12', <eof></eof>	~
	~	"'0.001''"	'0.001',Error Token '	'0.001',Error Token '	~
	~	"'0.001'''"	'0.001''', <eof></eof>	'0.001''', <eof></eof>	~

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00. Accounting for previous tries, this gives **0.90/1.00**.

