1. HL.jj

Create a JavaCC file called HL.jj that JavaCC can use to generate a scanner for HL. This scanner will be graded by a program, so it is important that you follow the specifications below carefully.

2Cc (which will automatically include 2De)

- HL is case sensitive except for the keywords which are case-insensitive.
- HL allows inline comments in the C++ style: anything after a "//" up to the end of the line it occurs on is a comment.   Your scanner should remove all comments and whitespace.
- HL contains the following keywords: "DO ", "FOR", "IN", "WHILE ", "IF", "THEN", "ELIF", "ELSE", "FI", "FUNCTION", "RETURN", "END", "PRINT", "PRINTLN", "VAR". Each of these should be a separate token with the same name as the string.
- In the table underneath, each of the following additional tokens is defined by the string below it:

Token	LESS	LESSEQ	GREATER	GREATEQ	EQUAL	NOTEQ	ISIN	NOTIN
String	<	<=	>	>=	==	!=	=in	!in
Token	PLUS	MINUS	TIMES	DIV	MOD	AND	VBAR	NOT
String	+	-	*	/	%	&	|	!
Token	ASSIGN	LROUND	RROUND	LCURLY	RCURLY	LSQUARE	RSQUARE	
String	=	(	)	{	}	[	]	
Token	COMMA	SEMICOL	TRUE	FALSE				
String	,	;	#1	#0				

2Ba:

- IDNUM tokens are strings of 1 or more letters and digits which must start with a lower-case letter.
- IDSET tokens are strings of 1 or more letters and digits which must start with an upper-case letter.
- IDBOOL tokens are strings that start with a '#' and a letter, followed by 0 or more letters and digits.
- NUMBER tokens are strings of 1 or more digits.
- STRING tokens are strings of keyboard characters delimited by double-quotes (").

2Aa:

- HL also allows nested comments which are delimited by "/'" and "'/". Remove nested comments.

2. Token.java, NumberToken.java, IdNumToken.java, IdSetToken.java, IdBoolToken.java, andStringToken.java

2Ba:

- Following the instructions above the constructor for Token, extend the Token class for NUMBER, IDNUM, IDSET, IDBOOL, and STRING tokens. Call them NumberToken.java , IdNumToken.java, IdSetToken.java, IdBoolToken.java, and StringToken respectively.
- These Token classes contain an additional public variable called value which contains the value of the token. Value is of type Integer in NumberToken, and String in all the others. The value in the StringToken should be the string inside the surrounding double quotes.
- Override the getValue() method of Token in these subclasses. This returns the value of the token if there is one, or null if there isn't. Note that the only tokens which have a value are NUMBER, IDNUM, IDSET, IDBOOL, and STRING.
- Override the toString() method of Token in the subclasses to be the toString() method of the value. This will ensure that the values are properly printed. For example, the leading 0s of integers will be removed.

2Aa:

- – As in C and Java, in HL "\" is the escape character. HL has 4 string escapes: \n for newline, \t for tab, \" for
  " and \\ for \. Change the value stored in the StringToken to replace the escaped characters by the real
  characters they represent. Now HL strings will be properly printed.