Exercise 10: Scanning Character Constants

Implement the method readCharCon(t) according to its specification in Chapter 2. This method is called by next() to recognize a character constant (e.g., 'x'). The token type charCon should be stored in t.kind and the numeric value of the character constant in t.numVal. If the character constant is invalid (e.g., '', 'abc'), an error message should be printed.

Solution

When readCharCon() is called, ch contains the opening apostrophe. readCharCon() must now continue reading up to the closing apostrophe (or up to the end of the line) and store the contents of the character constant in a char array lex. This array is then analyzed and its numerical character value is stored in t.numVal. If the character constant is incorrect, an error message is printed, but t.kind is still set to charCon so as not to "confuse" the parser.

```
private static char[] lex = new char[128];

private static void readCharCon (Token t) {
    int i = 0;
    t.kind = charCon;
    nextCh();
    while (ch != '\'' && ch != '\n' && ch != eofCh) {     // read character constant
        lex[i++] = ch; nextCh();
    }
    if (ch == '\n' || ch == eofCh) {
        Parser.error("invalid character constant");
    } else if (i == 1 && lex[0] != '\\') {                // analyze character constant
        t.numVal = lex[0];
    } else if (i == 2 && lex[0] == '\\') {
        if      (lex[1] == 'r') t.numVal = '\r';
        else if (lex[1] == 'n') t.numVal = '\n';
        else if (lex[1] == 't') t.numVal = '\t';
        else Parser.error("invalid character constant");
    } else Parser.error("invalid character constant");
    if (ch == '\'') nextCh();
}
```

To report an error, we use the method Parser.error() here, which will also be used by the parser to print error messages including the line and column number of the current token. However, as long as the parser does not yet exist, error messages can simply be printed to the console (e.g., System.out.println("invalid character constant");).