Pragmatic programming

1. Tokens made from characters
   1. Concatenation
   2. Alteration
   3. Kleine closure (repeating symbols)
2. Regular grammar- from regular expressions
3. Context free language- set of strings that can be defined with recursion
4. Regex
   1. Null character
   2. Empty string
   3. To adjascent regex
   4. Two regex sperated by |
5. Sentential form- all derivation maxed
6. Derivations- a reduction from a production
7. Association- read left to right
8. Precedence- PEMDAS
9. Scanner groups things into tokens, removes comments
   1. Saves work for parser
10. Can represent dfa using tables or nested switch statement
11. Parser
    1. Calls scanner to put code into tree