GRAMMAR

```
File = Header Sheet

Header = Index Title (?: Composer|Length|Meter|Tempo|(Voice Voice+)){n=number
of incidences} Key

Index = [X:] [/d] [/r|/n]
Title = [T:] [/S]+ [/r|/n]
Composer = [C:] [/S]+ [/r|/n]
Length = [L:] [1][//][/d] [/r|/n]
Meter = [M:] ([/d][//][/d])|[C] [/r|/n]
Tempo = [Q:] [/d]* [/r|/n]
Key = [C|D|E|F|G|A|B][m|M]? [/r|/n]
Voice = [V:] [/S]+ [/r|/n]

Sheet = (Meter|Repeat)*

Meter = Notes LINE
Notes = (Single|Chord|Tuplet|Triplet|Rest)*
Single = ACCIDENTAL? NOTE OCTAVE* Length?
Length = ([/d]*[///]*[/d]*)+
Chord = LEFTBRA Single+ RIGHTBRA
Tuplet = LEFTPAR [2] Single{2}
Triplet = LEFTPAR [3] Single{3}
Rest = [z] Length?

Repeat = COLON Notes LINE Altend? Notes COLON LINE (Altend Notes LINE)?
Altend = LEFTBRA [/d]


LINE: |
ACCIDENTAL: ^ or _
NOTE: a-gA-G
OCTAVE: ' or ,
LEFTBRA: [
RIGHTBRA: ]
LEFTPAR: (
COLON: :
```

Data types:

Class Token
Class Header
Interface Music

Token:
-Types: {NOTE, LINE, ACCIDENTAL, OCTAVE, FRACTION, LEFTBRA, RIGHTBRA, LEFTPAR, REST}
-fields: String value, String originalString

Header:
-fields: String title, String composer, float length, float meter, int tempo, String key, String[] voice

Music:
-Class Sheet
-Class Meter
+-fields: Notes notes
-Class Repeat
+-fields: Meter rep1, Meter rep2, Meter alter
-Class Notes
+-fields: Token[] tokens
-Class Single
+-fields: Token token, boolean hasAcc, boolean hasOctave, Length length
-Class Length
+-fields: float value
-Class Chord
+-fields: Single[] singles
-Class Tuplet
+-fields: Single[2] singles
-Class Triplet
+-fields: Single[3] singles
-Class Rest
+-fields: float length
-Class Altend
+-fields: int order