## **Project 2: Music21**

For this project, we were tasked with exploring Python libraries and finding one that interested us. Then we had to create a code sample that explores the different ways that the library can be used. We had to do a total of 3 different libraries per group, so we each took the reins on one, and helped each other with input and ideas for the ones our group members primarily worked on.

The library I have chosen is music21. Music21 is “a toolkit for computer-aided music analysis and computational musicology”. This library can be used to create music through code and analyze music as well. You can create scores, as well as turn them into different forms such as audio, midi’s and more. It also can be used to answer questions about music, such as how many times a certain chord progression is used, or which band might’ve used a chord progression first. The possibilities are endless.

Since this library can go very in depth with what I can do, I decided to create 2 different scores with it. One being more simple that shows the fundamentals of adding notes, and one that goes more in depth of adding multiple parts to a score and creating chords.

There’s a lot of functions included in the library. In fact, there are over 50 chapters of different functions that you can use in the library. Because of this, I am going to go over the main ones that I used in my program:

**stream.Score()** - creates a score for the music to be written in

**score.append()** - adds to the score

**note.Note()** - creates a note

**meter.TimeSignature()** - sets the time signature for the score

**key.KeySignature()** - sets the key signature for the score

**tempo.MetronomeMark()** - sets the tempo for the score, including the bpm, the feel, and what note is the base tempo.

**stream.Part()** - creates a part for the score.

**n.duration.type** - sets the length of the notes

**s.insert()** - inserts the part where we want to in the score (ex: (0, part1)

**s.show()** - opens/shows the score in MuseScore or GarageBand depending on the type of show, for example, s.show(‘midi’) opens a MIDI in a GarageBand.

Overall, this project gave us great insight into the capabilities of libraries in Python, and the fact that there are tens of thousands of them is mindblowing. The possibilities are truly endless and I am definitely going to look into more libraries in my spare time.

Link to Library Documentation: <https://web.mit.edu/music21/doc/usersGuide/usersGuide_99_Table_of_Contents.html>