*ClefHandler* Source Code.

import abjad

class ClefHandler:

def \_\_init\_\_(

self,

clef=None,

ottava\_shelf=None,

add\_ottavas=False,

):

self.clef = clef

self.ottava\_shelf = ottava\_shelf

self.add\_ottavas = add\_ottavas

def \_\_call\_\_(self, selections):

return self.add\_clef(selections)

def add\_clef(self, selections):

for run in abjad.select(selections).runs():

ties = abjad.select(run).logical\_ties(pitched=True)

if self.clef != None:

abjad.attach(abjad.Clef(self.clef), ties[0][0])

if self.add\_ottavas == True:

self.\_add\_ottavas(selections)

return selections

def \_add\_ottavas(self, selections):

if self.clef == 'treble':

if self.ottava\_shelf != None:

shelf = self.ottava\_shelf

for tie in abjad.select(selections).logical\_ties():

for pitch in abjad.inspect(tie[0]).pitches():

if pitch > shelf:

abjad.ottava(tie)

else:

shelf = 36

for tie in abjad.select(selections).logical\_ties(pitched=True):

for pitch in abjad.inspect(tie[0]).pitches():

if pitch > shelf:

abjad.ottava(tie)

if self.clef == 'alto':

if self.ottava\_shelf != None:

shelf = self.ottava\_shelf

for tie in abjad.select(selections).logical\_ties(pitched=True):

for pitch in abjad.inspect(tie).pitches():

if pitch > shelf:

abjad.ottava(tie)

else:

shelf = 13

for tie in abjad.select(selections).logical\_ties(pitched=True):

for pitch in abjad.inspect(tie[0]).pitches():

if pitch > shelf:

abjad.ottava(tie)

if self.clef == 'varC':

if self.ottava\_shelf != None:

shelf = self.ottava\_shelf

for tie in abjad.select(selections).logical\_ties(pitched=True):

for pitch in abjad.inspect(tie[0]).pitches():

if pitch > shelf:

abjad.ottava(tie)

else:

shelf = 13

for tie in abjad.select(selections).logical\_ties(pitched=True):

for pitch in abjad.inspect(tie[0]).pitches():

if pitch > shelf:

abjad.ottava(tie)

if self.clef == 'tenor':

if self.ottava\_shelf != None:

shelf = self.ottava\_shelf

for tie in abjad.select(selections).logical\_ties(pitched=True):

for pitch in abjad.inspect(tie[0]).pitches():

if pitch > shelf:

abjad.ottava(tie)

else:

shelf = 10

for tie in abjad.select(selections).logical\_ties(pitched=True):

for pitch in abjad.inspect(tie[0]).pitches():

if pitch > shelf:

abjad.ottava(tie)

if self.clef == 'tenorvarC':

if self.ottava\_shelf != None:

shelf = self.ottava\_shelf

for tie in abjad.select(selections).logical\_ties(pitched=True):

for pitch in abjad.inspect(tie[0]).pitches():

if pitch > shelf:

abjad.ottava(tie)

else:

shelf = 10

for tie in abjad.select(selections).logical\_ties(pitched=True):

for pitch in abjad.inspect(tie[0]).pitches():

if pitch > shelf:

abjad.ottava(tie)

if self.clef == 'bass':

if self.ottava\_shelf != None:

shelf = self.ottava\_shelf

for tie in abjad.select(selections).logical\_ties(pitched=True):

for pitch in abjad.inspect(tie[0]).pitches():

if pitch > shelf:

abjad.ottava(tie)

else:

shelf = 3

for tie in abjad.select(selections).logical\_ties(pitched=True):

for pitch in abjad.inspect(tie[0]).pitches():

if pitch > shelf:

abjad.ottava(tie)

return(selections)