**Key**

Forms

Variable Creation/Definition

Conditional Clause

Loop

File Operations

Logging Operations

Function Creation

Developer Comment

*This is the code to display the form used at the beginning of the script. To permanently change these values they must be changed in the script. It is also possible to disregard the form entirely if you set the variables in the script only. For example: sound\_directory = ‘C:/Users/Some\_User/Some\_directory’*

form Analyze pitch maxima from labeled segments in files

comment Directory of sound files

text sound\_directory C:\Users\Dave\Downloads\SampleFilesPraat\sound\

sentence Sound\_file\_extension .wav

comment Directory of TextGrid files

text textGrid\_directory C:\Users\Dave\Downloads\SampleFilesPraat\grid\

sentence TextGrid\_file\_extension .TextGrid

comment Full path of the resulting text file:

text resultfile C:\Users\Dave\Downloads\SampleFilesPraat\results\output.csv

comment Which tier do you want to analyze?

sentence Tier ASD

comment Pitch analysis parameters

positive Time\_step 0.01

positive Minimum\_pitch\_(Hz) 100

positive Maximum\_pitch\_(Hz) 700

endform

*the author does a great job with this explanation*

# Here, you make a listing of all the sound files in a directory. # The example gets file names ending with ".wav" from D:\tmp\

Create Strings as file list... list 'sound\_directory$'\*'sound\_file\_extension$'

numberOfFiles = Get number of strings

*if the file can be read by the script, it must exist and therefore may contain data that should not be removed*

if fileReadable (resultfile$)

pause The result file 'resultfile$' already exists! Do you want to overwrite it?

filedelete 'resultfile$'

endif

*I modified this section to include titles for the additional column headers. In addition I changed the output to comma separated values in order to accommodate working with the data in excel. Feel free to change the names of any of the fields (if you remove any of them, be sure to remove the corresponding title at the starred line) \*\*\*\*\**

titleline$ = "Filename,Segment label,Maximum pitch (Hz),Minimum pitch (Hz),mean,median,stdev,altqb,dur'newline$'"

fileappend "'resultfile$'" 'titleline$'

# Go through all the sound files, one by one:

for ifile to numberOfFiles

filename$ = Get string... ifile

# A sound file is opened from the listing:

Read from file... 'sound\_directory$''filename$'

# Starting from here, you can add everything that should be

# repeated for every sound file that was opened:

soundname$ = selected$ ("Sound", 1)

To Pitch... time\_step minimum\_pitch maximum\_pitch

# Open a TextGrid by the same name:

gridfile$ = "'textGrid\_directory$''soundname$''textGrid\_file\_extension$'"

if fileReadable (gridfile$)

Read from file... 'gridfile$'

# Find the tier number that has the label given in the form:

call GetTier 'tier$' tier

numberOfIntervals = Get number of intervals... tier

# Pass through all intervals in the selected tier:

for interval to numberOfIntervals

label$ = Get label of interval... tier interval

if label$ <> ""

# if the interval has an unempty label, get its start and end:

start = Get starting point... tier interval

end = Get end point... tier interval

dur = (start + end) / 2

# get the Pitch maximum at that interval

select Pitch 'soundname$'

pitchmax = Get maximum... start end Hertz Parabolic

printline 'pitchmax'

# get the Pitch minimum at that interval

select Pitch 'soundname$'

pitchmin = Get minimum... start end Hertz Parabolic

printline 'pitchmin'

mean = Get mean... start end Hertz

median = Get quantile... start end 0.5 Hertz

stdev = Get standard deviation... start end Hertz

altqb = Get quantile... start end 0.0764 Hertz

baseline = mean - (1.43 \* stdev)

# Save result to text file:

resultline$ = "'soundname$','label$','pitchmax','pitchmin','mean','median','stdev','altqb','dur''newline$'" # \*\*\*\*\*

fileappend "'resultfile$'" 'resultline$'

select TextGrid 'soundname$'

endif

endfor

# Remove the TextGrid object from the object list

select TextGrid 'soundname$'

Remove

endif

# Remove the temporary objects from the object list

select Sound 'soundname$'

plus Pitch 'soundname$'

Remove

select Strings list

# and go on with the next sound file!

endfor

Remove

#-------------

# This procedure finds the number of a tier that has a given label.

procedure GetTier name$ variable$

numberOfTiers = Get number of tiers

itier = 1

repeat

tier$ = Get tier name... itier

itier = itier + 1

until tier$ = name$ or itier > numberOfTiers

if tier$ <> name$

'variable$' = 0

else

'variable$' = itier - 1

endif

if 'variable$' = 0

exit The tier called 'name$' is missing from the file 'soundname$'!

endif

endproc