```python
#PSP_travelTimeAnalysis
from __future__ import print_function, division

from PSP_readDataFile import *
from PSP_TTwaterContent import *
from PSP_TTplot import *
from PSP_travelTime import computeTravelTime

import sys
if sys.version_info >= (3,0):
    from tkinter import *                           #3.x
    from tkinter.filedialog import askopenfilename
    from tkinter.messagebox import showerror
else:
    from Tkinter import *                           #2.7
    from tkFileDialog import askopenfilename
    from tkMessageBox import showerror

isDataLoaded = False
waterTemperature = 20
liquidPermittivity = getLiquidPermittivity(waterTemperature)

mainWindow = Tk()
mainWindow.title("TDRPy")
mainWindow.geometry("%dx%d" % (300, 650))

headerNrStr = StringVar()
vpStr = StringVar()
probeLenghtStr = StringVar()
windowBeginStr = StringVar()
windowWidthStr = StringVar()
probleHandleStr = StringVar()
handlePermittivityStr = StringVar()
point0XStr = StringVar()
point1XStr = StringVar()
point2XStr = StringVar()
ttStr = StringVar()
bulkPermittivityStr = StringVar()
waterTemperatureStr = StringVar()
bulkDensityStr = StringVar()
solidPermittivityStr = StringVar()
liquidPermittivityStr = StringVar()
geometricParStr = StringVar()
wcToppStr = StringVar()
wcMalickiStr = StringVar()
wcMixModelStr = StringVar()

if sys.version_info >= (3, 0):
    waterTempLabel = Label(mainWindow, text="Water temp. [\u00B0C]")
else:
    waterTempLabel = Label(mainWindow, text="Water temp. [C]")
waterTempLabel.place(x=80, y=275)
waterTempWidget = Entry(mainWindow, width = 6, textvariable = waterTemperatureStr)
waterTempWidget.place(x=200, y=275)

def getEpsilonLabel():
    if sys.version_info >= (3, 0):
        return Label(mainWindow, font = "helvetica 12", text="\u03F5")
    else:
        return Label(mainWindow, font = "helvetica 10", text="e")
```

Make a pop-up window for input and output





```python

def checkTWater(event):
    global waterTemperature, liquidPermittivity
    if (event.widget == waterTempWidget):
        waterTemperature = float(waterTemperatureStr.get())
        liquidPermittivity = getLiquidPermittivity(waterTemperature)
        liquidPermittivityStr.set(format(liquidPermittivity,".3f"))

def importData():
    nrHeaderValues = int(headerNrStr.get())
    fileName = askopenfilename()
    if (fileName != ""):
        global isDataLoaded, waveFormNrpoints
        x, isFileOk = readDataFile(fileName,nrHeaderValues,'\t', False)
        if (not isFileOk):
            showerror("Wrong file", "Error reading row nr." + str(x))
            return(False)
        if len(x) == 1:
            data = x[0,:]
        else:
            data = x[:,0]
        tt.reflecCoeff = tt.normalizeVector(data)
        print("number of values:", len(data))
        isDataLoaded = True

def ComputeTT():
    if not isDataLoaded:
        showerror("Warning", "Data not loaded")
        return
    #read parameters
    vp = float(vpStr.get())
  #fraction of speed of light [-]
    probeLenght = float(probeLenghtStr.get())
    windowBegin = float(windowBeginStr.get())
    windowWidth = float(windowWidthStr.get())
    probeHandle = float(probleHandleStr.get())
    handlePermittivity = float(handlePermittivityStr.get())
    bulkDensity = float(bulkDensityStr.get())
    solidPermittivity = float(solidPermittivityStr.get())
    geomParameter = float(geometricParStr.get())

    #compute
    nrPoints = len(tt.reflecCoeff)
    tt.WF_parameters(vp, probeHandle, windowBegin, windowWidth, nrPoints)
    if not computeTravelTime(probeHandle, handlePermittivity, vp):
        showerror("Warning", "Wrong data, header or parameter")
        return
    travelTime = tt.p2.x - tt.p1.x
    bulkPermittivity = getBulkPermittivity(probeLenght, travelTime, vp)
    wcTopp = getWaterContentTopp(bulkPermittivity)
    wcMalicki = getWaterContentMalicki(bulkPermittivity, bulkDensity)
    wcMixModel = getWaterContentMixModel(bulkPermittivity, bulkDensity,
                        solidPermittivity, liquidPermittivity, geomParameter)
    #print results
    x0 = tt.p0.x * (1E09)
    point0XStr.set(format(x0, '.3f'))
    x1 = tt.p1.x * (1E09)
    point1XStr.set(format(x1, '.3f'))
    x2 = tt.p2.x * (1E09)
    point2XStr.set(format(x2, '.3f'))
```

```
121     ttStr.set(format(travelTime * (1E09), '.3f'))
122     bulkPermittivityStr.set(format(bulkPermittivity, '.2f'))
123     wcToppStr.set(format(wcTopp,".3f"))
124     wcMalickiStr.set(format(wcMalicki,".3f"))
125     wcMixModelStr.set(format(wcMixModel,".3f"))
126 
127     #graph
128     cleanDisplay()
129     drawWaveForm()
130     drawRegressionLines()
131     showDisplay()
132 
133 def main():
134     vpStr.set(0.99)
135     probeLenghtStr.set(0.15)
136     windowBeginStr.set(0.)
137     windowWidthStr.set(5.)
138     probleHandleStr.set(0.108)
139     handlePermittivityStr.set(1.7)
140     point0XStr.set(0)
141     point1XStr.set(0)
142     point2XStr.set(0)
143     ttStr.set(0)
144     bulkPermittivityStr.set(0)
145     bulkDensityStr.set(1350)
146     waterTemperatureStr.set(waterTemperature)
147     solidPermittivityStr.set(4.0)
148     liquidPermittivityStr.set(format(liquidPermittivity, ".3f"))
149     geometricParStr.set(0.5)
150     wcToppStr.set(0)
151     wcMalickiStr.set(0)
152     wcMixModelStr.set(0)
153 
154     buttonImport = Button(mainWindow, text="Import data", command=importData)
155     buttonImport.place(x=5, y=15)
156 
157     headerLabel = Label(mainWindow, text="Header values nr:")
158     headerLabel.place(x=110, y=15)
159     headerWidget = Entry(mainWindow, width = 3, textvariable = headerNrStr)
160     headerWidget.place(x=200, y=15)
161     headerWidget.insert(0, "8")
162     headerWidget.pack
163 
164     dataFormatLabel = Label(mainWindow, text="Settings" , font = "helvetica 10
    bold", fg="red")
165     dataFormatLabel.place(x=20, y=50)
166 
167     computeTTButton = Button(mainWindow, text="Compute", command=ComputeTT)
168     computeTTButton.place(x=5, y=80)
169 
170     vpLabel = Label(mainWindow, text="Vp [-]")
171     vpLabel.place(x=90, y=80)
172     vpWidget = Entry(mainWindow, width = 6, textvariable = vpStr)
173     vpWidget.place(x=200, y=80)
174 
175     probeLenghtLabel = Label(mainWindow, text="Probe length [m]")
176     probeLenghtLabel.place(x=80, y=105)
177     probeLenghtWidget = Entry(mainWindow, width = 6, textvariable = probeLenghtStr)
178     probeLenghtWidget.place(x=200, y=105)
179 
```

Import data ボタンを押すと
importData を実行

Compute ボタンを押すと
ComputeTT を実行





```
180     winBeginLabel = Label(mainWindow, text="Window begin [m]")
181     winBeginLabel.place(x=80, y=130)
182     winBeginWidget = Entry(mainWindow, width = 6, textvariable = windowBeginStr)
183     winBeginWidget.place(x=200, y=130)
184 
185     winwidthLabel = Label(mainWindow, text="Window width [m]")
186     winwidthLabel.place(x=80, y=155)
187     winwidthWidget = Entry(mainWindow, width = 6, textvariable = windowWidthStr)
188     winwidthWidget.place(x=200, y=155)
189 
190     probeHandleLabel = Label(mainWindow, text="Probe handle [m]")
191     probeHandleLabel.place(x=80, y=180)
192     probeHandleWidget = Entry(mainWindow, width = 6, textvariable =
    probleHandleStr)
193     probeHandleWidget.place(x=200, y=180)
194 
195     epsilonLabel = getEpsilonLabel()
196     epsilonLabel.place(x=80, y=200)
197     epsilonLabel = Label(mainWindow, text="handle")
198     epsilonLabel.place(x=90, y=206)
199     permittivityWidget = Entry(mainWindow, width = 6, textvariable =
    handlePermittivityStr)
200     permittivityWidget.place(x=200, y=200)
201 
202     SoilParameterLabel = Label(mainWindow, text="Soil parameters" , font =
    "helvetica 10 bold", fg="red")
203     SoilParameterLabel.place(x=20, y=225)
204 
205     bulkDensityLabel = Label(mainWindow, text="Bulk density [m^3 kg]")
206     bulkDensityLabel.place(x=80, y=250)
207     bulkDensityWidget = Entry(mainWindow, width = 6, textvariable = bulkDensityStr)
208     bulkDensityWidget.place(x=200, y=250)
209 
210     epsilon2Label = getEpsilonLabel()
211     epsilon2Label.place(x=80, y=295)
212     epsilon2Label = Label(mainWindow, text="liquid")
213     epsilon2Label.place(x=90, y=300)
214     liquidPermittivityLabel = Label(mainWindow, textvariable =
    liquidPermittivityStr)
215     liquidPermittivityLabel.place(x=200, y=300)
216 
217     epsilon3Label = getEpsilonLabel()
218     epsilon3Label.place(x=80, y=320)
219     epsilon3Label = Label(mainWindow, text="solid")
220     epsilon3Label.place(x=90, y=325)
221     solidPermittivityWidget = Entry(mainWindow, width = 6, textvariable =
    solidPermittivityStr)
222     solidPermittivityWidget.place(x=200, y=325)
223 
224     geometricParLabel = Label(mainWindow, text="alpha (geom. param.)")
225     geometricParLabel.place(x=80, y=350)
226     geometricParWidget = Entry(mainWindow, width = 6, textvariable =
    geometricParStr)
227     geometricParWidget.place(x=200, y=350)
228 
229     ttResultsLabel = Label(mainWindow, text="Travel Time results" , font =
    "helvetica 10 bold", fg="blue")
230     ttResultsLabel.place(x=20, y=375)
231 
232     point0Label = Label(mainWindow, text="point 0 x [ns]")
```

```
    point0Label.place(x=80, y=400)
    point0Widget = Label(mainWindow, width = 6, textvariable = point0XStr)
    point0Widget.place(x=200, y=400)

    point1Label = Label(mainWindow, text="point 1 x [ns]")
    point1Label.place(x=80, y=425)
    point1Widget = Label(mainWindow, width = 6, textvariable = point1XStr)
    point1Widget.place(x=200, y=425)

    point2Label = Label(mainWindow, text="point 2 x [ns]")
    point2Label.place(x=80, y=450)
    point2Widget = Label(mainWindow, width = 6, textvariable = point2XStr)
    point2Widget.place(x=200, y=450)

    ttLabel = Label(mainWindow, text="Travel Time [ns]")
    ttLabel.place(x=80, y=475)
    ttWidget = Label(mainWindow, width = 6, textvariable = ttStr)
    ttWidget.place(x=200, y=475)

    bulkPermittivityLabel = Label(mainWindow, text="Bulk permittivity")
    bulkPermittivityLabel.place(x=80, y=500)
    bulkPermittivityWidget = Label(mainWindow, width = 6, textvariable =
bulkPermittivityStr)
    bulkPermittivityWidget.place(x=200, y=500)

    wcLabel = Label(mainWindow, text="Water Content" , font = "helvetica 10 bold",
fg="blue")
    wcLabel.place(x=20, y=525)

    ToppLabel = Label(mainWindow, text="Topp")
    ToppLabel.place(x=80, y=550)
    ToppWidget = Label(mainWindow, width = 6, textvariable = wcToppStr)
    ToppWidget.place(x=200, y=550)

    MalickiLabel = Label(mainWindow, text="Malicki")
    MalickiLabel.place(x=80, y=575)
    MalickiWidget = Label(mainWindow, width = 6, textvariable = wcMalickiStr)
    MalickiWidget.place(x=200, y=575)

    dielecMixModelLabel = Label(mainWindow, text="Diel. mix model")
    dielecMixModelLabel.place(x=80, y=600)
    dielecMixModelWidget = Label(mainWindow, width = 6, textvariable =
wcMixModelStr)
    dielecMixModelWidget.place(x=200, y=600)

    mainWindow.bind("<Leave>", checkTWater)
    mainWindow.protocol("WM_DELETE_WINDOW", mainWindow.destroy)

    mainWindow.mainloop()
main()
```