```
#<--28/02/2023
#<--Output of Graphics (Dawson's Code)
#<--author: Pedro Henrique Rocha de Andrade

#importing needed library
import numpy as np #calc
import pandas as pd #to read data
import matplotlib.pyplot as plt #plot graphics


#defining the files that will be used
local  = ".\Documentos\FEBRACE\English Version\Codes"
dir_cat = local + "Results/"

files = []

#range of masses from 1 to 1
files.append('zh_1to1_b0_z_updated')
#files.append('zh_1to1_b0.5_z_updated')
#files.append('zh_1to1_b1_z_updated')

#range of masses from 1 to 3
files.append('zh_1to3_b0_z_updated')
#files.append('zh_1to3_b0.5_z_updated') #impact parameter 0.5
#files.append('zh_1to3_b1_z_updated') #impact parameter 1

#range of masses from 1 to 10
files.append('zh_1to10_b0_z_updated')
#files.append('zh_1to10_b0.5_z_updated')
#files.append('zh_1to10_b1_z_updated')



#Naming the outputs (ZuHone)
out1 = []

out1.append('zh_1to1_b0_z.png')
out1.append('zh_1to3_b0_z.png')
out1.append('zh_1to10_b0_z.png')

#Naming the ZuHone vs Dawson comparison outputs
out2 = []
out2.append('dw_1to1_b0_z.png')
out2.append('dw_1to3_b0_z.png')
out2.append('dw_1to10_b0_z.png')

#Naming the ZuHone vs Dawson comparison outputs (with the Zoom at the focused area)
out3 = []
out3.append('dw_1to1_b0_z_zoom.png')
out3.append('dw_1to3_b0_z_zoom.png')
out3.append('dw_1to10_b0_z_zoom.png')


#Defining the time of collisions (smallest separation in parsec)
CT=[1.32,1.20,1.04]

#Defining the graphics' title
names=['1 to 1','1 to 3','1 to 10']


for i in range(0, len(files)):

        print(files[i])

        filename = dir_cat + files[i] + '.txt'
        df = pd.read_csv(filename, sep='\t')

# plotting the initial graphs
```

```
        plt.scatter(df['age']-CT[i],df['sep.kpc'],c='red')#setting the graph type to points (XY
dispersion) in red color
        plt.plot([0, 0], [0, 3100], 'k-') #Plotting the graphic
        plt.title(names[i]) #putting the title
        plt.xlabel('Time (Ganos)') #x axis title
        plt.ylabel('Distance (kpc)') #y axis title
        #plt.show() #showing the graphic
        plt.savefig(dir_cat + out1[i]) #saving the graphic
        plt.clf() #clearing

#comparison
        plt.scatter(df['age']-CT[i],df['sep.Mpc'],c='red') #ZuHone
        plt.scatter(df['TSC0'],df['d.proj.out'],c='blue') #Dawson

        y_error = [df['TSC0.lower'], df['TSC0.upper']]  #margin of error (Dawson)
        plt.errorbar(df['TSC0'],df['d.proj.out'], yerr=y_error, fmt='o')  #plot the error bar on
the charts

#Final graphics
        plt.title(names[i]) #putting the title
        plt.xlabel('Time (Ganos)') #identifying x axis
        plt.ylabel('Distance (Mpc)') #identifying y axis
        #plt.show() #showing graphics
        plt.savefig(dir_cat + out2[i]) #saving
        plt.clf() #clearing

#comparison with zoom
        plt.scatter(df['age']-CT[i],df['sep.Mpc'],c='red') #ZuHone
        plt.scatter(df['TSC0'],df['d.proj.out'],c='blue') #Dawson

        y_error = [df['TSC0.lower'], df['TSC0.upper']]  #margin of error (Dawson)
        plt.errorbar(df['TSC0'],df['d.proj.out'], yerr=y_error, fmt='o') #plot the error bar on
the charts

#final graphics (w/ zoom)
        plt.title(names[i]) #putting the title
        plt.xlabel('Time (Ganos)') #identifying x axis
        plt.ylabel('Distance (Mpc)') #identifying y axis
        plt.ylim(0,3) #zooming
        plt.xlim(0,1) #zooming
        #plt.show() #showing the graph
        plt.savefig(dir_cat + out3[i]) #saving
        plt.clf() #clearing
```