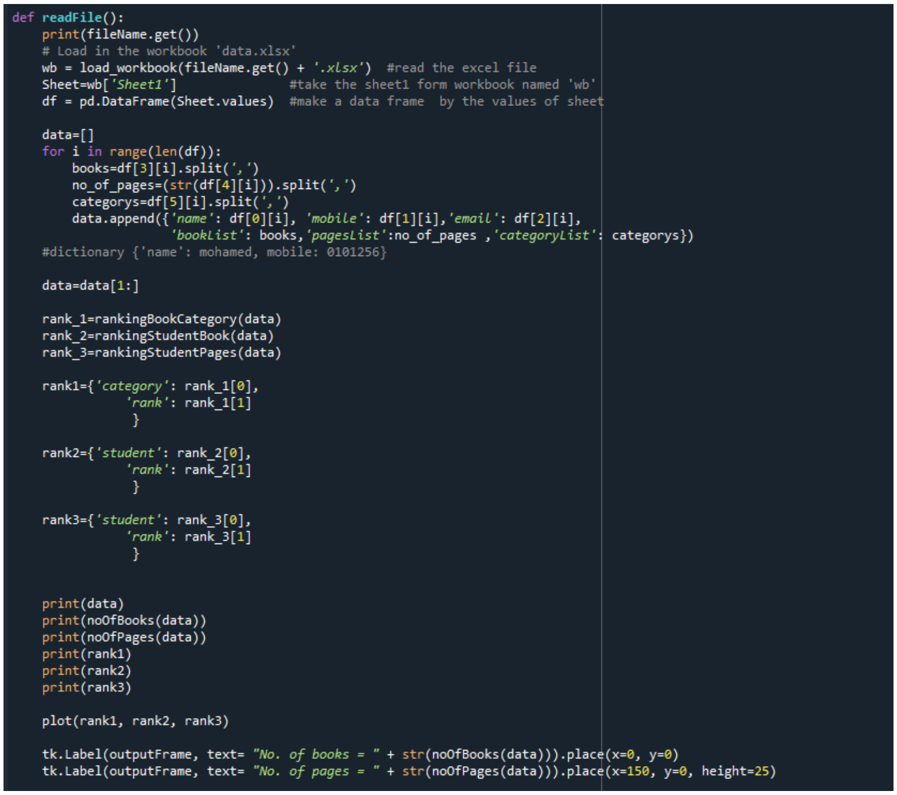
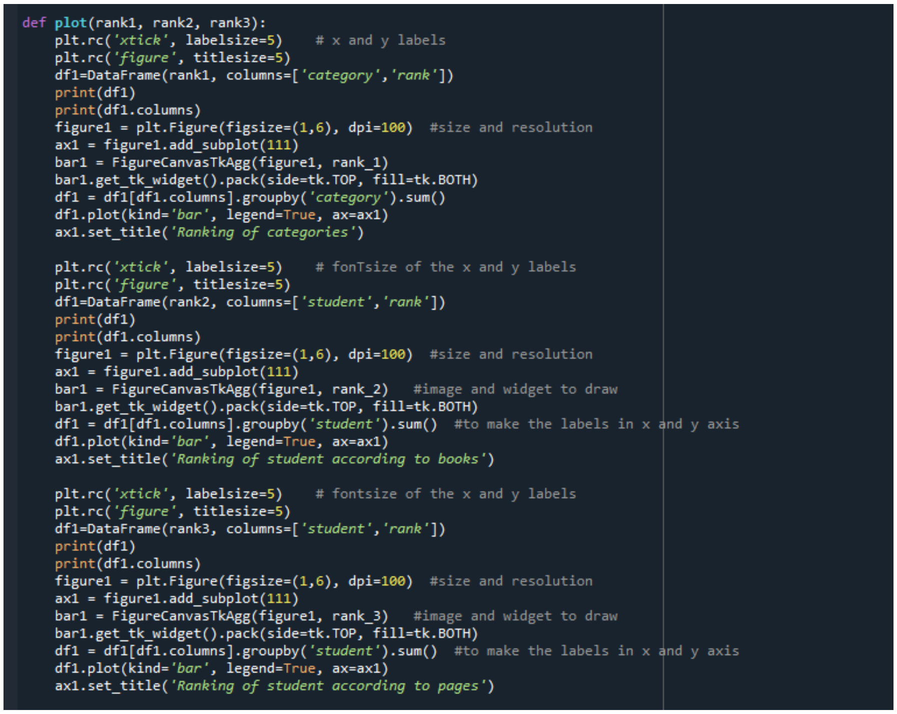
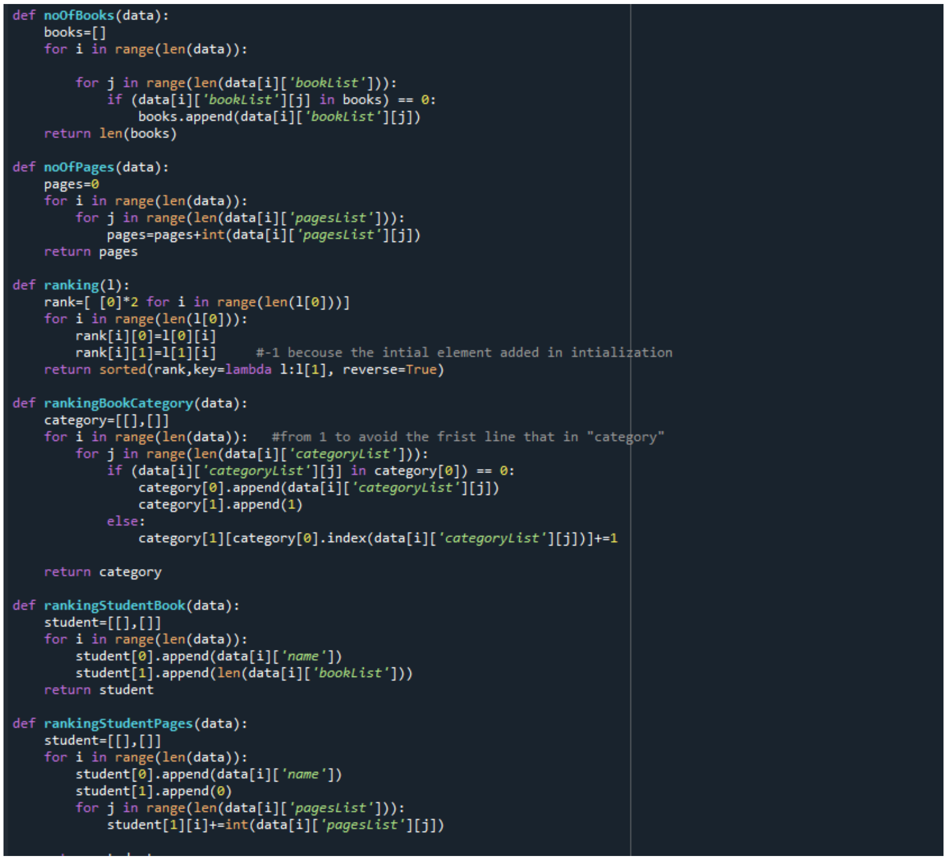


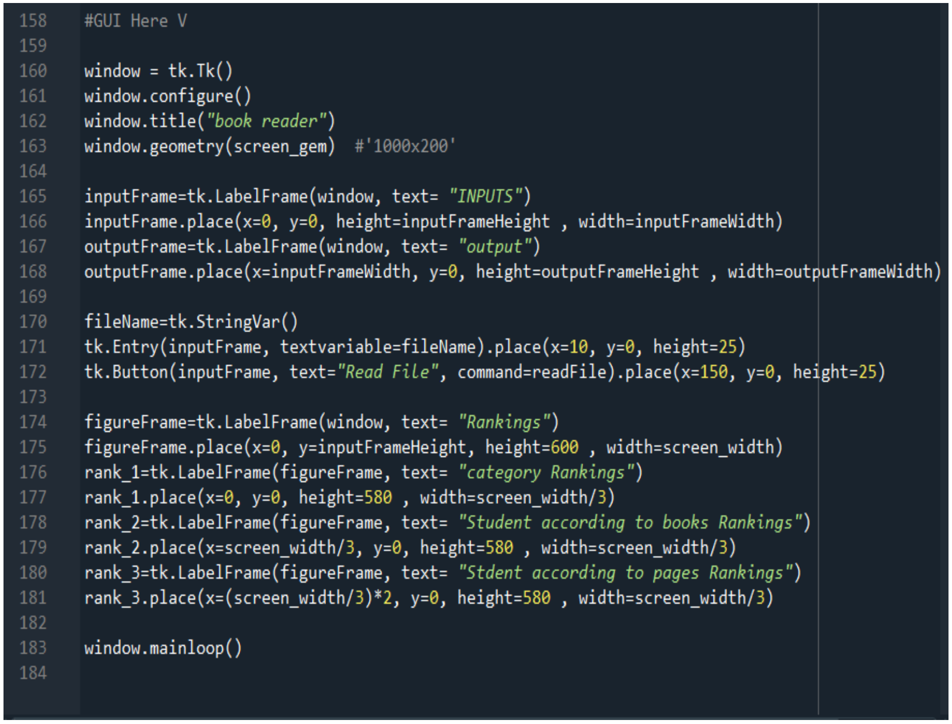
```
def readFile():
    print(fileName.get())
    # Load in the workbook 'data.xlsx'
    wb = load_workbook(fileName.get() + '.xlsx')  #read the excel file
    Sheet=wb['Sheet1']               #take the sheet1 form workbook named 'wb'
    df = pd.DataFrame(Sheet.values)  #make a data frame  by the values of sheet

    data=[]
    for i in range(len(df)):
        books=df[3][i].split(',')
        no_of_pages=(str(df[4][i])).split(',')
        categorys=df[5][i].split(',')
        data.append({'name': df[0][i], 'mobile': df[1][i],'email': df[2][i],
                     'bookList': books,'pagesList':no_of_pages ,'categoryList': categorys})
    #dictionary {'name': mohamed, mobile: 0101256}

    data=data[1:]

    rank_1=rankingBookCategory(data)
    rank_2=rankingStudentBook(data)
    rank_3=rankingStudentPages(data)

    rank1={'category': rank_1[0],
           'rank': rank_1[1]
            }

    rank2={'student': rank_2[0],
           'rank': rank_2[1]
            }

    rank3={'student': rank_3[0],
           'rank': rank_3[1]
            }


    print(data)
    print(noOfBooks(data))
    print(noOfPages(data))
    print(rank1)
    print(rank2)
    print(rank3)

    plot(rank1, rank2, rank3)

    tk.Label(outputFrame, text= "No. of books = " + str(noOfBooks(data))).place(x=0, y=0)
    tk.Label(outputFrame, text= "No. of pages = " + str(noOfPages(data))).place(x=150, y=0, height=25)
```



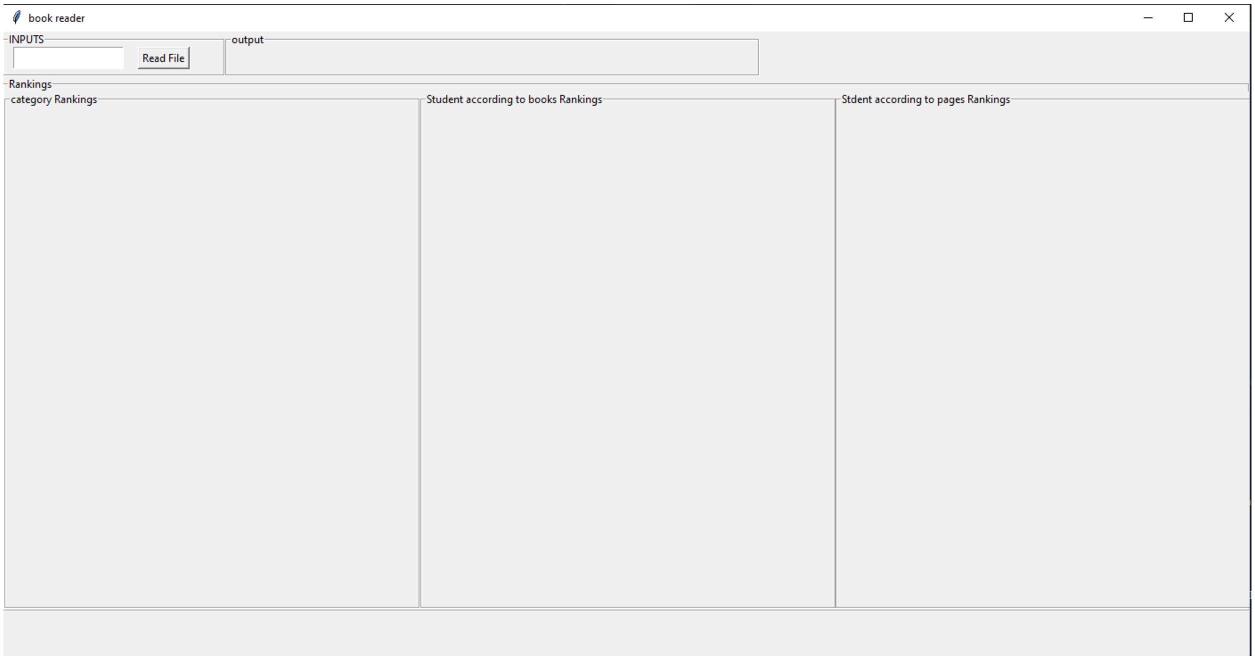
```
def plot(rank1, rank2, rank3):
    plt.rc('xtick', labelsize=5)    # x and y labels
    plt.rc('figure', titlesize=5)
    df1=DataFrame(rank1, columns=['category','rank'])
    print(df1)
    print(df1.columns)
    figure1 = plt.Figure(figsize=(1,6), dpi=100)  #size and resolution
    ax1 = figure1.add_subplot(111)
    bar1 = FigureCanvasTkAgg(figure1, rank_1)
    bar1.get_tk_widget().pack(side=tk.TOP, fill=tk.BOTH)
    df1 = df1[df1.columns].groupby('category').sum()
    df1.plot(kind='bar', legend=True, ax=ax1)
    ax1.set_title('Ranking of categories')

    plt.rc('xtick', labelsize=5)    # fonTsize of the x and y labels
    plt.rc('figure', titlesize=5)
    df1=DataFrame(rank2, columns=['student','rank'])
    print(df1)
    print(df1.columns)
    figure1 = plt.Figure(figsize=(1,6), dpi=100)  #size and resolution
    ax1 = figure1.add_subplot(111)
    bar1 = FigureCanvasTkAgg(figure1, rank_2)   #image and widget to draw
    bar1.get_tk_widget().pack(side=tk.TOP, fill=tk.BOTH)
    df1 = df1[df1.columns].groupby('student').sum()  #to make the labels in x and y axis
    df1.plot(kind='bar', legend=True, ax=ax1)
    ax1.set_title('Ranking of student according to books')

    plt.rc('xtick', labelsize=5)    # fontsize of the x and y labels
    plt.rc('figure', titlesize=5)
    df1=DataFrame(rank3, columns=['student','rank'])
    print(df1)
    print(df1.columns)
    figure1 = plt.Figure(figsize=(1,6), dpi=100)  #size and resolution
    ax1 = figure1.add_subplot(111)
    bar1 = FigureCanvasTkAgg(figure1, rank_3)   #image and widget to draw
    bar1.get_tk_widget().pack(side=tk.TOP, fill=tk.BOTH)
    df1 = df1[df1.columns].groupby('student').sum()  #to make the labels in x and y axis
    df1.plot(kind='bar', legend=True, ax=ax1)
    ax1.set_title('Ranking of student according to pages')
```

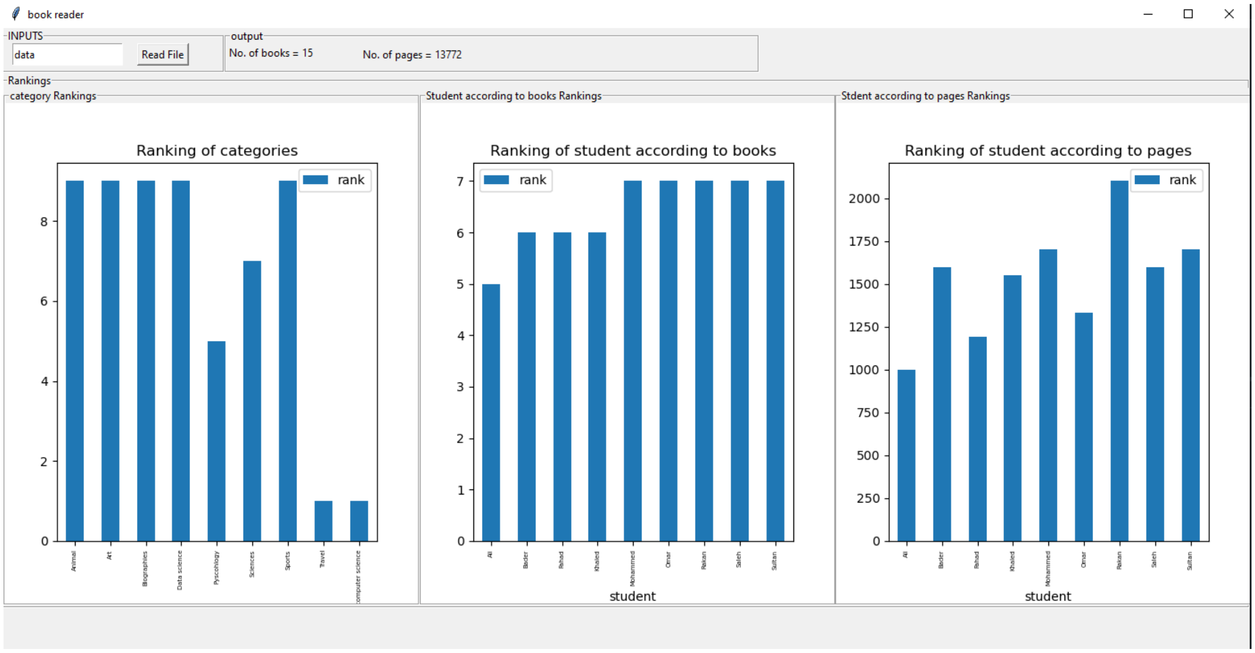


```python
def noOfBooks(data):
    books=[]
    for i in range(len(data)):

        for j in range(len(data[i]['bookList'])):
            if (data[i]['bookList'][j] in books) == 0:
                books.append(data[i]['bookList'][j])
    return len(books)

def noOfPages(data):
    pages=0
    for i in range(len(data)):
        for j in range(len(data[i]['pagesList'])):
            pages=pages+int(data[i]['pagesList'][j])
    return pages

def ranking(l):
    rank=[ [0]*2 for i in range(len(l[0]))]
    for i in range(len(l[0])):
        rank[i][0]=l[0][i]
        rank[i][1]=l[1][i]     #-1 becouse the intial element added in intialization
    return sorted(rank,key=lambda l:l[1], reverse=True)

def rankingBookCategory(data):
    category=[[],[]]
    for i in range(len(data)):   #from 1 to avoid the frist line that in "category"
        for j in range(len(data[i]['categoryList'])):
            if (data[i]['categoryList'][j] in category[0]) == 0:
                category[0].append(data[i]['categoryList'][j])
                category[1].append(1)
            else:
                category[1][category[0].index(data[i]['categoryList'][j])]+=1

    return category

def rankingStudentBook(data):
    student=[[],[]]
    for i in range(len(data)):
        student[0].append(data[i]['name'])
        student[1].append(len(data[i]['bookList']))
    return student

def rankingStudentPages(data):
    student=[[],[]]
    for i in range(len(data)):
        student[0].append(data[i]['name'])
        student[1].append(0)
        for j in range(len(data[i]['pagesList'])):
            student[1][i]+=int(data[i]['pagesList'][j])
```



```python
#GUI Here V

window = tk.Tk()
window.configure()
window.title("book reader")
window.geometry(screen_gem)  #'1000x200'

inputFrame=tk.LabelFrame(window, text= "INPUTS")
inputFrame.place(x=0, y=0, height=inputFrameHeight , width=inputFrameWidth)
outputFrame=tk.LabelFrame(window, text= "output")
outputFrame.place(x=inputFrameWidth, y=0, height=outputFrameHeight , width=outputFrameWidth)

fileName=tk.StringVar()
tk.Entry(inputFrame, textvariable=fileName).place(x=10, y=0, height=25)
tk.Button(inputFrame, text="Read File", command=readFile).place(x=150, y=0, height=25)

figureFrame=tk.LabelFrame(window, text= "Rankings")
figureFrame.place(x=0, y=inputFrameHeight, height=600 , width=screen_width)
rank_1=tk.LabelFrame(figureFrame, text= "category Rankings")
rank_1.place(x=0, y=0, height=580 , width=screen_width/3)
rank_2=tk.LabelFrame(figureFrame, text= "Student according to books Rankings")
rank_2.place(x=screen_width/3, y=0, height=580 , width=screen_width/3)
rank_3=tk.LabelFrame(figureFrame, text= "Stdent according to pages Rankings")
rank_3.place(x=(screen_width/3)*2, y=0, height=580 , width=screen_width/3)

window.mainloop()
```


book reader
INPUTS
Read File
output
Rankings
category Rankings
Student according to books Rankings
Stdent according to pages Rankings



data

```
No. of books = 15        No. of pages = 13772
```

Language feature	Class	Line No.
If/else	noOfBooks	111
	rankingBookCategory	133-137
For loop	readFile	26
	noOfBooks	108
	noOfBooks	110
	noOfPages	117
	noOfPages	118
	ranking	124
	rankingBookCategory	131
	rankingBookCategory	132
	rankingStudentBook	143
	rankingStudentPages	150
	rankingStudentPages	153