YouTube Video Downloader

Team - UwU, Group - 1
Roll no. - 11, Ishaant Kumar Singh, Reg no. - 12203987
Roll no. - 16,Muhammad Yousuf Khan, Reg no. - 12204116

The Youtube downloader project is a python project. The object of this project is to download any type of video in a fast and easy way from Youtube in your device and watch them offline.

In this python project, user has to copy the Youtube video URL that they want to download and simply paste that URL in the ‘paste link here’ section and click on the ‘get info’ button to fetch the information like Title, Author, Length of the video, Views about the video. Then click on the ‘Browse’ button to select the destination of the video where you want to download and then finally click on the download button and it will start downloading the video. When video downloading finishes you can check the folder you selected for downloading the video.

```
from tkinter import *
import tkinter as tk
from tkinter import ttk
from pytube import YouTube
from tkinter import messagebox, filedialog
from PIL import ImageTk, Image

root = Tk()
```

```
mainlogo=ImageTk.PhotoImage(Image.open("mainlogo.png"))
```

```
def browsePath():
    download_Directory = filedialog.askdirectory(title="Save Video")
    pathBox.configure(text=download_Directory)
    print(pathBox.cget("text"))
```

```
list1 = []
```

```
def getInfo():
```

```
    messagebox.showinfo("Alert","Getting your Video Info \n Please wait")
```

```
    global list1
```

```
    ytLink = entrylink.get()
    yt = YouTube(ytLink)
    resolution =[int(i.split("p")[0]) for i in (list(dict.fromkeys([i.resolution for i in
yt.streams if i.resolution])))]
```

```
    resolution.sort()
```

```
    for i in (resolution):
        a = str(i)
        b = "p"
        c = a + b
        list1.append(c)
```

```
    print(list1)
    videoResolution.config(values=list1)
```

```
    vtitle = "Title : "+ yt.title
    vauthor = "Author : " +  yt.author
    vdate = "Published date : " + yt.publish_date.strftime("%Y-%m-%d")
    vviews = "Number of views : " , yt.views
    vlenght = "Length of video : " , yt.length , "seconds"
```

```
    infolabel1.config(text=vtitle)
    infolabel2.config(text=vauthor)
    infolabel3.config(text=vdate)
    infolabel4.config(text=vviews)
    infolabel5.config(text=vlenght)
```

```
def download():
```

```
    messagebox.showinfo("Downloading your Video","It might take some time \n depending on your internet connection \n Window will stop working due to threading \n Please wait ")

    global yt

    ytLink = entrylink.get()
    ytresolution = videoResolution.get()
    downloadPath = pathBox.cget("text")

    print(ytresolution)
    print(ytLink)


    YouTube(ytLink).streams.filter(res=ytresolution).first().download(downloadPath)


mainlog=Label(image=mainlogo)
mainlog.grid(column=1,row=1)

entrylink = ttk.Entry(root,width=21,)
entrylink.grid(column=1, row=2)

button2 = ttk.Button( root ,width=10,text = "get info" , command=getInfo )
button2.grid(column=1, row=3,ipadx=10, ipady=10)


infolabel1 = Label(root, text=' ')
infolabel1.grid(column=1, row=4)

infolabel2 = Label(root, text=' ')
infolabel2.grid(column=1, row=5)

infolabel3 = Label(root, text=' ')
infolabel3.grid(column=1, row=6)

infolabel4 = Label(root, text=' ')
infolabel4.grid(column=1, row=7)

infolabel5 = Label(root, text=' ')
infolabel5.grid(column=1, row=8)


videoResolution = ttk.Combobox(root, values=list1)
videoResolution.grid(column=1, row=10)


pathBox = Label(root,text="Choose path :/")
pathBox.grid(column=1, row=11)

pathBtn = ttk.Button( root , text='Browse' , command=browsePath)
pathBtn.grid(column=2, row=11,ipadx=10, ipady=10)


button1 = ttk.Button( root , text = "Download" , command=download)
button1.grid(column=1, row=12,ipadx=10, ipady=10)


canvas= Canvas(root, width= 50, height= 50)
canvas.grid()
```

```
icon= PhotoImage(file = 'icon.png')
root.iconphoto(False,icon)
root.title('YtHUB')
```

```
root.geometry("400x500")
root.resizable(False, False)
```

```
root.mainloop()
```