

פרויקט בתכנות מונחה אובייקטים 'א':**מה הפרויקט עושה:**

המשתמש מכניס שם של שיר, התוכנה מחזירה לו רשימה של חמישה שירים מהם הוא בוחר את השיר שהוא רוצה. לאחר בחירת השיר נפתח השיר והקליפ בחלון חדש באתר יוטיוב.

חבילות שהורדנו:

בשביל להוריד את כל החבילות הרלוונטיות עשינו את השלבים הבאים:
 name of the package → '+' → python interpreter → setting → fill
 רשימה של החבילות ומה כל חבילה עושה -

- א. import webbrowser** – פתיחת מידע מדפדפן אינטרנט.
- ב. import requests** – ספרייה לעבודה עם HTTP. מפשט את התהליך של קבלת וקריאת הנתונים מהאינטרנט מול כתיבת הקוד ב python .
- ג. from bs4 import BeautifulSoup** - ספרייה שמקלה על העבודה בין האינטרנט לפיתון, אפשר להבין בעזרתה איך בנוי האתר אינטרנט ולגשת למידע הרלוונטי בפיתון לפי כותרות וכו'.
- ד. from selenium import webdriver** - מייצג את העבודה מול הדפדפן (פתיחת דפדפן, גלישה לאתרים וכו'). חבילת selenium של פיתון משמשת לביצוע בדיקות אוטומטיות של אתרים כי היא שולטת בדפדפן ומאפשרת לכוון את ההתנהגות שלו. בעזרת הספרייה הזו התגברנו על הקושי לגשת ישירות לאתרי אינטרנט של גוגל. בעצם אנחנו פותחות דפדפן אינטרנט חדש "שעוקף" את הקשיים של הגישה ישירה לאתרי גוגל.
- ה. from selenium.webdriver.common.by import By** - בודקים האם האתר אינטרנט שאליו אנחנו מפנים בכלל קיים.
- ו. import time** – חבילה שבעזרתה מגדירים את הזמן עד שחלון האינטרנט יפתח "זמן הדיליי".
- ז. import tkinter as tk** – הורדת GUI בעזרו אנו מעצבות את החלון שנפתח למשתמש, כלומר העיצוב של הממשק בין התוכנה למשתמש.
- ח. from tkinter import Tk, Button, Label, Entry, StringVar, Listbox, END, SINGLE** - פירוט של הפונקציות ב GUI שבהם השתמשנו.

הסבר כללי על הקוד:

- **שורות 1-8:** החבילות בהן השתמשנו. (מפורט למעלה)
- **שורות 11-37:** שמות המשתנים המופיעים בקוד (הסבר על כל אחד בהמשך)
- **שורות 40-125:** מחלקה שיצרנו ובה מספר פונקציות האחראיות על כל התהליך של חיפוש סרטונים ביוטיוב והצגת התוצאות למשתמש באמצעות ממשק גרפי. (פירוט על כל פונקציה בהמשך).
 המחלקה YouTubeDisplayProgram נועדה ליצור ממשק גרפי (GUI) שבו המשתמש יכול להזין שם של שיר, לבצע חיפוש ביוטיוב, ולהציג רשימה של תוצאות חיפוש רלוונטיות. התוצאה המוצגת היא רשימה של עד 5 שירים, וכאשר המשתמש בוחר אחד מהם, ייפתח הקליפ ביוטיוב בחלון חדש.
- **שורות 128-130:** הפעלת פעולות המחלקה בעת הרצת הקוד.

פירוט של שמות המשתנים:

SLEEP_TIME = הזמן שמחכים עד שנפתח החלון לאחר בחירת השיר הרצוי
YOUTUBE_URL = כתובת המייצגת מיקום יוטיוב באינטרנט
SEARCH_URL = הכתובת של יוטיוב באינטרנט בה נחפש את השיר
TITLE_REQUEST_FROM_USER = שם השיר אותו המשתמש מכניס
WINDOW_OPEN_TITLE = שם (כותרת) החלונית של GUI שנפתח כשמריצים את התוכנית
WINDOW_FRAME_SIZE = גודל החלונית של GUI שנפתחת
WINDOW_BACKGROUND_COLOR = צבע החלונית של GUI שנפתחת
INPUT_FRAME_COLOR = צבע מסגרת החלונית בה המשתמש מכניס את שם השיר
TITLE_FONT = גופן הכותרת בחלונית
TITLE_SIZE = גודל הכותרת בחלונית
INPUT_FONT = גופן הקלט שהמשתמש מכניס (שם השיר)
INPUT_FONT_SIZE = גודל הגופן של הקלט
INPUT_BOX_SIZE = גודל הלשונית בה המשתמש כותב את הקלט
BUTTON_LABEL = הכפתור עליו המשתמש לוחץ לאחר כתיבת הקלט
BUTTON_FONT = גופן הכפתור
BUTTON_FONT_SIZE = גודל גופן הכפתור
BUTTON_BACKGROUND = צבע הרקע של הכפתור
BUTTON_FONT_COLOR = צבע הגופן של הכפתור
OPTIONS_FONT = הגופן של 5 האופציות של השירים לאחר הכנסת הקלט ולחיצה על כפתור "ENTER"
OPTIONS_FONT_SIZE = גודל הגופן של האופציות המוצעות
OPTIONS_WINDOW_SIZE = גודל החלונית שבה רואים את האופציות המוצעות
OPTION_WINDOW_HIGH_SIZE = הגובה המקסימלי של חלונית האופציות המוצעות

פירוט של הפונקציות בהן השתמשנו:

א. `def get_html_content(search_url)`

המשתנה `search_url` מכיל את הכתובת URL של החיפוש ביוטיוב עליו עושים את החיפוש של הערך שהמשתמש הכניס.

שימוש ב `selenium`:

- מפעילים דפדפן כרום במצב `headless` – מצב שבו אין ממשק גרפי פתוח למשתמש ובכך אנו ניגשים לתוצאות.
- המשתנה `driver` : שם משתמש אליו טוענים את כתובת הURL שקלטנו מהמשתמש.
- `time.sleep` : מגדיר כמה זמן מחכים עד שהתוכן יטען.
- `element = driver.find_element(By.CSS_SELECTOR, "html")`
`html = element.get_attribute("outerHTML")`
- שתי השורות האלו מוצאות את מה שאנחנו מחפשים לפי הקלט של המשתמש ולוקחת את הURL המלא.
- לאחר מכן סוגרים את הדפדפן.

ב. : def update_list(html, listbox)

בעזרת הפונקציה הזו אנחנו מנתחים את כתובת ה URL ושואבים ממנה את המידע הרלוונטי לנו. בנוסף הפונקציה מעדכנת את listbox - משתנה שמציג את רשימת השירים שיוצגו למשתמש.

שימוש ב *BeautifulSoup* :

- משתמשים בחבילה זו בשביל "לפרק" את הכתובת URL ולקוחים ממנה את התגיות עם המידע הרלוונטי לנו - שמות השיר וקישורים לקליפ.
- list_of_options : מאתחלים רשימה ריקה שתכיל את האפשרויות של השירים והכתובות URL שלהם.
- לפני שטוענים את האופציות למשתנה listbox מוחקים את הנתונים הקודמים.
- בעזרת הלולאה רצים על כל תוצאות החיפוש. בכל פעם שהלולאה רצה אנחנו לוקחים את המידע הבא :
 1. שם השיר מתוך הכותרת : `txt = link.text.strip()`
 2. קישור לקליפ : `a = link.select("a")[0]['href']`
 3. בניית רשימה של השירים עם שם השיר מהכותרת והקישור URL : `lst_of_options.append({'name': txt, 'url': a})`
 4. הוספת שם השיר ל listbox : `listbox.insert(END, txt)`
- 5. לבסוף הפונקציה מחזירה את רשימת השירים עם כתובות ה URL.

ג. : def user_select(listbox, lst_of_options)

הפונקציה אחראית להפעיל פיזית את הבחירה של המשתמש. ראשית בודקים האם התבצעה בחירה ע"י המשתמש, במידה ולא התקבלה התוכנית מפסיקה לרוץ. לאחר מכן אנו שומרים את האינדקס של בחירת המשתמש - `index = listbox.curselection()[0]` - כעת בונים את הכתובת URL המלאה לפי בחירת המשתמש - `song_url = YOUTUBE_URL + lst_of_options[index]['url']` בשלב האחרון הפונקציה פותחת את הקישור של השיר - `webbrowser.open_new(song_url)`

ד. : def search_and_open(entry_song_name, listbox)

הפונקציה מבצעת חיפוש ביוטיוב על פי שם השיר שהמשתמש הזין, מעדכנת את Listbox עם תוצאות החיפוש, ומאפשרת למשתמש לבחור שיר מתוך הרשימה ולפתוח אותו בדפדפן.

- קבלת הקלט : `search = entry_song_name.get()`
- בניית כתובת URL מלאה : `search_url = SEARCH_URL + str(search)`
- שליחת הבקשה וקבלת ה HTML : `html = get_html_content(search_url)`
- עדכון תיבת הרשימה עם האופציות לבחירה : `lst_of_options = update_list(html, listbox)`
- בעזרת הפונקציה lambda אנו מבטיחים שהמשתמש יבחר באופציה אחת מכלל האופציות שיש ומחזירה את הקליפ.

`listbox.bind('<<ListboxSelect>>', lambda choose: user_select(listbox, lst_of_options))`

ה. : def create_gui()

הפונקציה אחראית על הממשק בין המשתמש למחשב היא יוצרת חלון אינטראקטיבי מול המשתמש. בחלון קולטים את השיר אותו רוצים לשמוע - מחזירים רשימה של חמש אופציות לבחירה, המשתמש בוחר אופציה אחת מתוכן ונפתח לו חלון חדש עם הקליפ מאתר יוטיוב.