קובץ הסבר לקוד:

קודם כל, הקוד שהבאתי הוא הקוד הבסיסי הנצרך בלבד בלי כל מיני הוספות שאני ודביר עשינו (למשל, פרסומות וכו' אין בגרסא הנוכחית שמצורפת)  
דבר שני, פיצלתי קצת את הקוד לצורך הנוחות.

הקובץ הראשון: pdfbooklet\_new.py

ובכן, לא התעמקתי בו יותר מדי, דביר שאל אותו מקוד פתוח בגיטהב. בגדול זו ספרייה שמכילה פונקציות נוחות ל"משחקים" על pdf.

הקובץ השני: ui\_settings.py

הקובץ מכיל את פונקציית UI(), הפונקציה הזו משמשת בתור מסך הפתיחה.  
היא מקבלת מהמשתמש את כל הנתונים הנדרשים בצורה נוחה ומסודרת. אין שום תחכום מיותר, בעיקר הנגשת התוכנה למשתמש.

הקובץ השלישי: pdf\_to\_tiny\_book\_2.1.2.py (נא לא להאשים אותי בשם הזה)

* main  
  הקובץ מכיל את הפונקציה הראשית (main) שקוראת לפונקציית UI ומעבירה את הנתונים מהמשתמש אל פונקציה אחרת שמבצעת את פעולת התוכנה של הכנת booklet לפי דרישות המשתמש. (השתמשנו בתהליכון לצורך שימוש פשוט יותר בקוד לצורך הוספת 'פיצ'רים' עתידיים)  
  הפונקציה הזאת גם מעלה הודעות בהתאם למצב התקדמות התוכנה. וכן גם אם המשתמש רוצה להשתמש בתוכנה פעם נוספת, יש אפשרות.

כעת אסביר מספר פונקציות בסיסיות, בהמשך אסביר את המהלך הכללי כפי היכולת.  
\* הערה: חשוב לציין שכל השמות שהמערכת "ממציאה" הם לפי פורמט מסודר ועקבי.

* extract\_num\_of\_pages(pdf\_path)  
  הפונקציה הזו, כשמה, מקבלת את מספר העמודים שיש לקובץ pdf מסוים (אם יש 50 עמודים בקובץ, אז הפונקציה תחזיר – 50)
* def split(path, name\_of\_split, sp, length, bind\_method='s')  
  מקבל מיקום של קובץ pdf שעליו מתבצעת הפוקנציה.  
  מקבלת את שם חדש בעבור הקובץ החדש שהפונקציה מייצרת.  
  חותכת מתוך הקובץ המקורי את העמודים שמתחילים באינדקס sp ועוד מספר עמודים לפי האורך שהתקבל. (במידה והאורך חורג החוצה מאשר גבולות הקובץ המקורי -> מוסיף דפים ריקים להשלמה)  
  bind – מוסיף דף לבן ריק בהתחלה ובסוף הקובץ, זה לצורך מאוחר יותר בתוכנה (האם הספר יהיה בהדבקה או בתפירה) – יש לי חשד סביר להניח שיש פה בעיה קטנה בקוד
* split\_Even\_Odd(path, name\_of\_split)  
  מפצל את הקובץ לשני קבצים. קובץ אחד מכיל את העמודים הזוגיים והשני מכיל את העמודים האי-זוגיים.  
  מוסיף עמודים ריקים בסוף לפי נוסחה מסויימת שלא הבנתי את ההיגיון שלה (אני חושב שהיא שורש של באג – כדאי לבדוק)  
  \* הערה: כנראה הנוסחה הזאת אמורה לגרום לכך שמספר הדפים יתחלק ב4 – יש צורך לתקן את השורה – נכון לעכשיו פשוט "מחקתי" את הנוסחה ושמתי 0 במקום. יש צורך לבדוק האם בכלל יש צורך בשורה הזו או לא.
* rotate(path, name\_of\_rotate, num\_rot=3)  
  כשמה כן היא, מסובבת את העמוד להיות לאורך או לרוחב. (משתנה לפי num\_rot שהתקבל)
* merge\_pdfs(paths, output)  
  מקבל רשימה של קבצי pdf, ומאחד אותם להיות קובץ אחד (חיבור ברצף, כך שהקובץ הראשון ראשון, השני שני וכן הלאה)
* merge\_sort\_pdfs(path1, path2, output)  
  מקבל שני קבצי pdf ומאחד ביניהם לסרוגין (כלומר, העמוד הראשון של הקובץ הראשון זה העמוד הראשון של הפלט, העמוד הראשון של הקובץ השני יהיה העמוד השני של הפלט, השני של הראשון יהיה השלישי של הפלט וכו')  
  \* הערה: ככל הנראה הפונקציה משמשת בעבור שני קבצים זהים בגודלם
* pile\_combine(file, path)  
  מקבל את הקובץ (path) מפצל אותו לשני חלקים שווים.  
  ואז מבצע עליהם את פונקציית איחוד לסירוגין (התיאור מופיע כאן לעיל)
* combineMethod(trash\_file, num)  
  מבצע את פונקציית pile\_combine פעמיים. אבל נותן שם שונה לכל קובץ שנוצר (אחד מקבל כותרת זוגי והשני כותרת אי זוגי)

עד כאן החלק הטכני והמשעמם של כל מיני פונקציות עזר לצורך ביצוע פונקציות על קבצי pdf.  
ייתכן מאוד שכדאי לשים אותן בקובץ אחר בתור ספרייה נפרדת.

את הפונקציות שעושות את עיקר הלוגיקה

* moreThan
* making\_the\_pdf

יותר מסובך להסביר במילים איך בדיוק זה עובד. בעז"ה אעבור על זה.

בעיקרון דביר הוא זה שעשה את החלק של הלוגיקה, אז באותה מידה שאני יכול לפרש את הקוד גם כולם יכולים.  
אשתדל להמשיך להוסיף הסברים אם יש צורך בהמשך