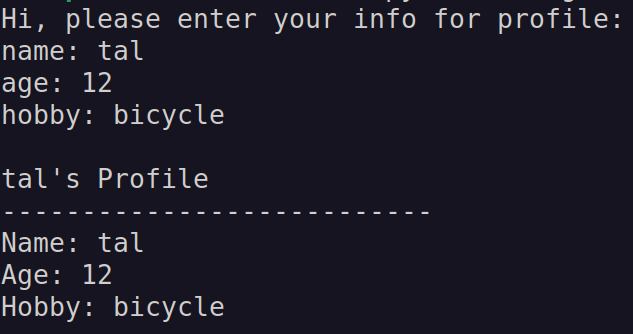
תרגילים בטכניקות ליבה

* משתנים ו datatypes
* תנאים ולולאות
* אוספים (רשימות, מילונים), מחרוזות (strings)
* פונקציות
* שימוש בספריות סטנדרטיות וחיצוניות

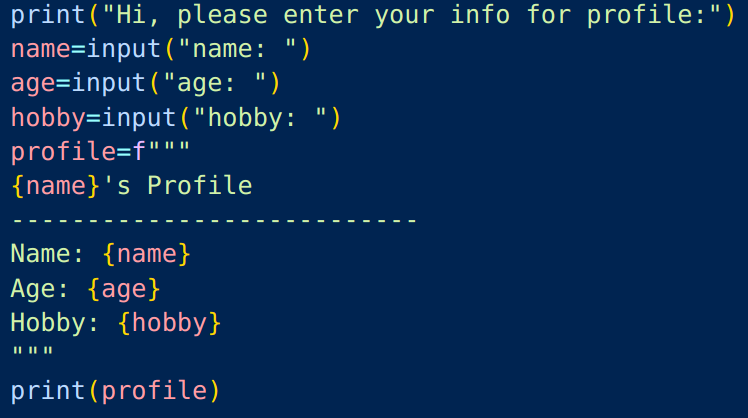
**תרגיל 1:**

כתבו תוכנית המקבלת מהמשתמש פרטים אישיים ומייצרת מחרוזת (string) עם פרופיל משתמש. הציגו את הפרופיל למשתמש.

דוגמת הרצה:



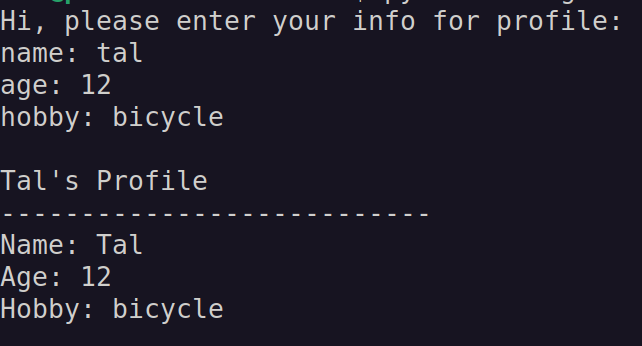
פתרון:



**תרגיל 2:**

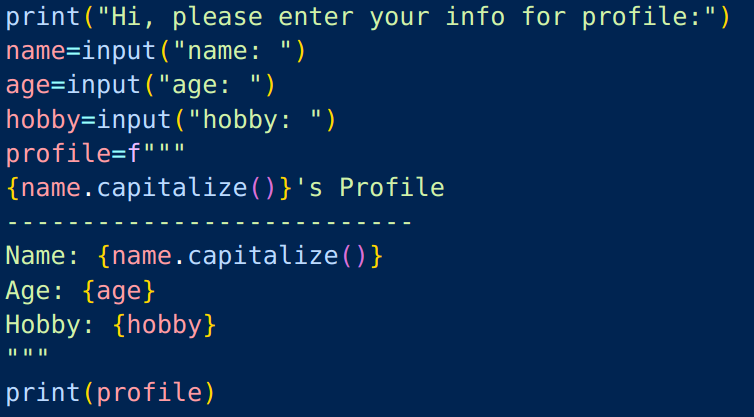
בתרגיל 1, הדפיסו את שם המשתמש בפרופיל עם אות גדולה.

דוגמת הרצה:



פתרון:

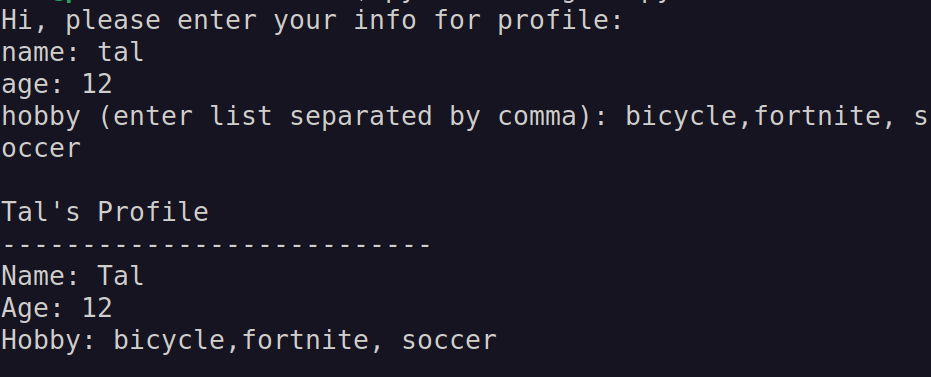
שימוש במתודה capitalize של string:



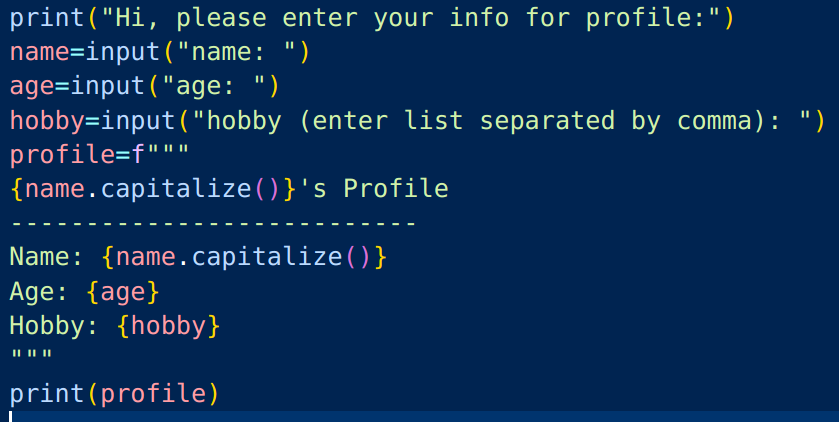
**תרגיל 3:**

בתרגיל הקודם, קבלו מהמשתמש רשימה של תחביבים והציגו אותה.

דוגמת הרצה:

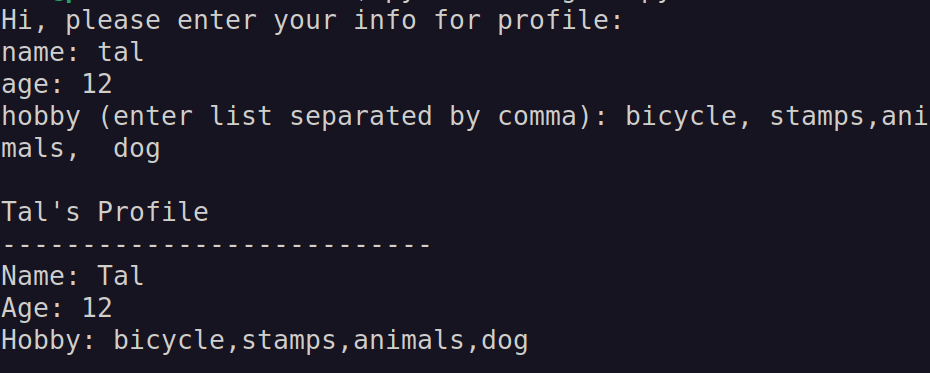


פתרון:



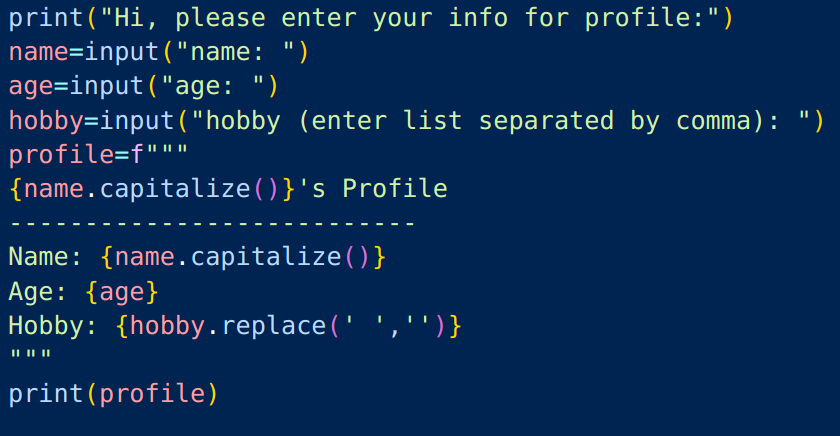
**תרגיל 4:**

בתרגיל הקודם, הסירו את הרווחים מרשימת התחביבים והציגו את הרשימה ללא רווחים:



פתרון:

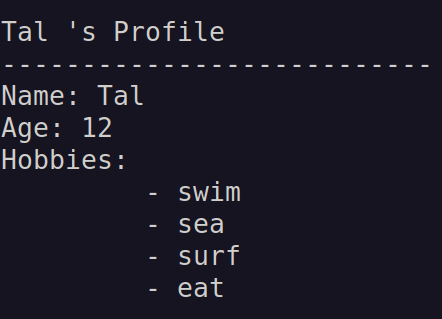
שימוש במתודה replace של string:



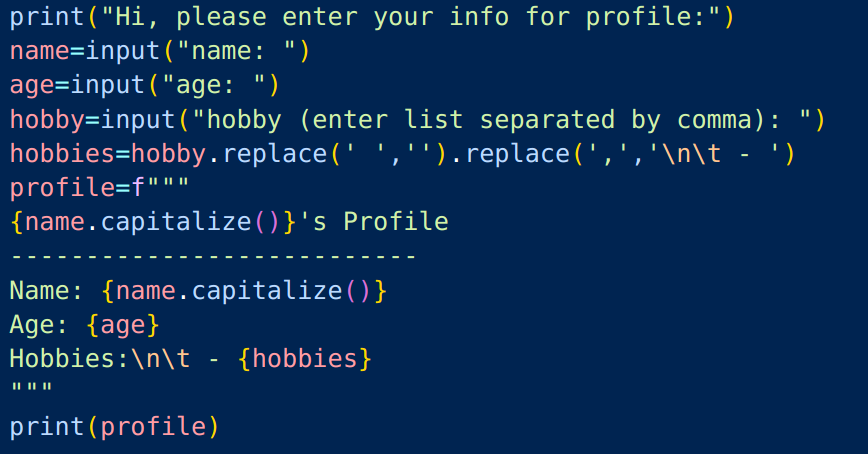
**תרגיל 5:**

בתרגיל הקודם, הציגו את רשימת התחביבים כתת-רשימה של הפרופיל.

דוגמת הרצה:



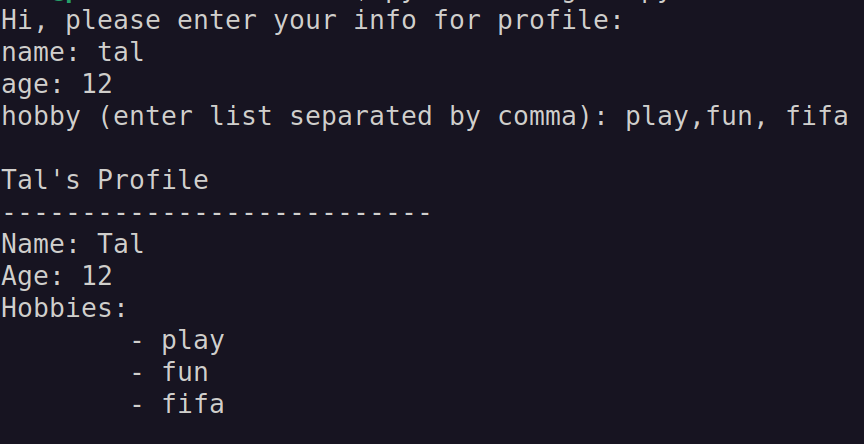
פתרון:



**תרגיל 6:**

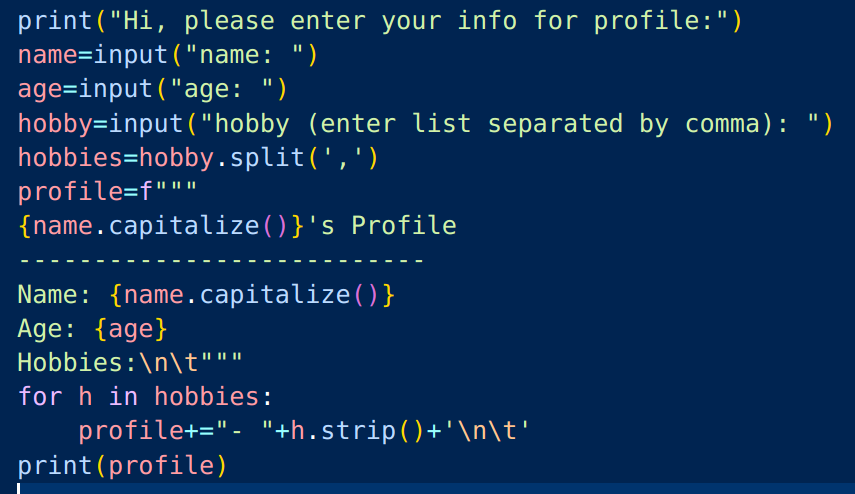
שימרו את התחביבים מהתרגיל הקודם ברשימה, והציגו אותה כמו בתרגיל הקודם.

דוגמת הרצה:



פתרון:

נשתמש במתודה split של string שמחזירה רשימה:



**תרגיל**

כתבו פונקציה המקבלת את השם ומחזירה True אם האורך לפחות 2 תווים, אחרת מחזירה False.

**תרגיל**

* הוסיפו לפונקציה פרמטרים של אורך מינימלי ומקסימלי בנוסף למחרוזת.
* בדקו אם אורך מינימלי 2 ומקסימום 7

**תרגיל**

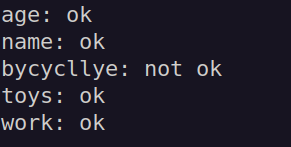
כתבו פונקציה המקבלת גיל ומחזירה True אם הוא בין 7-70, אחרת False.

**תרגיל 9**

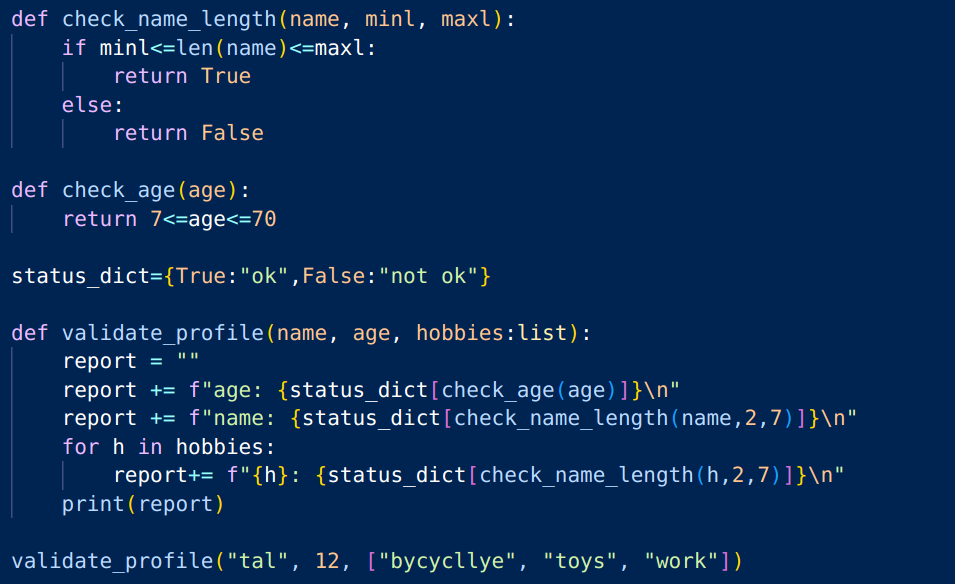
כתבו פונקציה המקבלת שם, גיל ותחביבים ובודקת את כולם לפי הדרישות בתרגילים הקודמים:

* מדפיסה דו"ח כישלון לכל פרמטר
* מחזירה את אלה שהצליחו ברשימה

דוגמת הרצה:



פתרון:



**תרגיל**

כתבו פונקציה המקבלת תחביב.

בודקת האם התחביב הוא string

אם כן, תחזיר True.

אם לא, תבקש string עד שתקבל ותחזיר אותו.

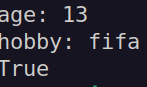
**תרגיל**

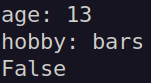
כתבו פונקציה המקבלת תחביב וגיל.

האם התחביב ברשימת תחביבים קבועה מראש לפי הגיל?

אם כן, תחזיר True.

אם לא, תחזיר False.





**תרגיל 7:**

כיתבו פונקציה המקבלת רשימת תחביבים ומחזירה מחרוזת עם רשימה עם פסיקים.

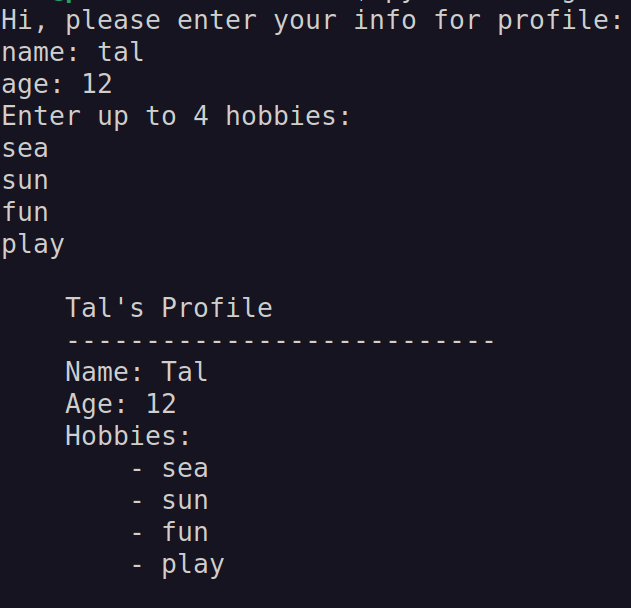
דוגמת הרצה: כמו בתרגיל הקודם

פתרון:

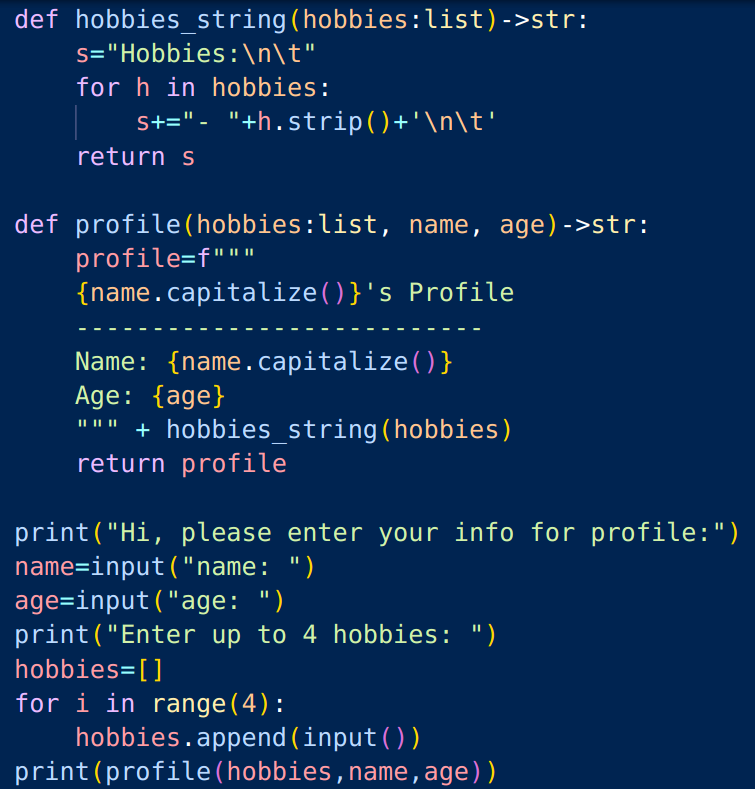
**תרגיל 9:**

במקום לבקש רשימת תחביבים, בקשו 4 תחביבים והוסיפו אחד אחד לרשימה.

דוגמת הרצה:



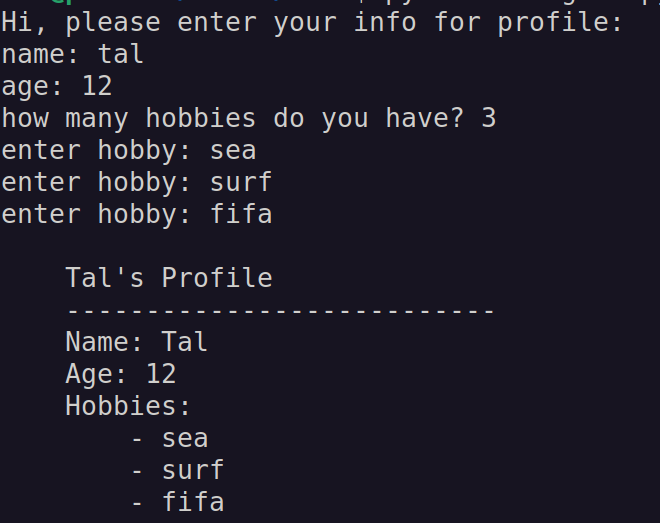
פתרון:



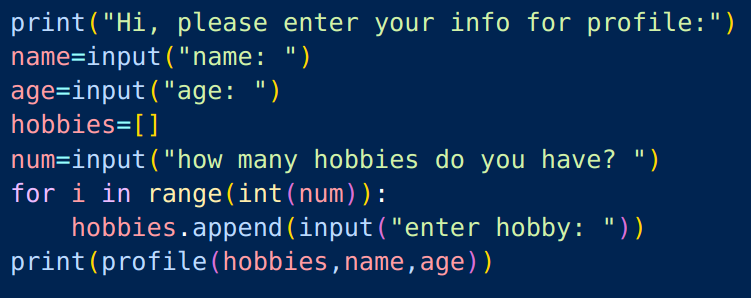
**תרגיל 10**

בתרגיל הקודם, קבלו את מספר התחביבים מהמשתמש.

דוגמת הרצה:



פתרון:



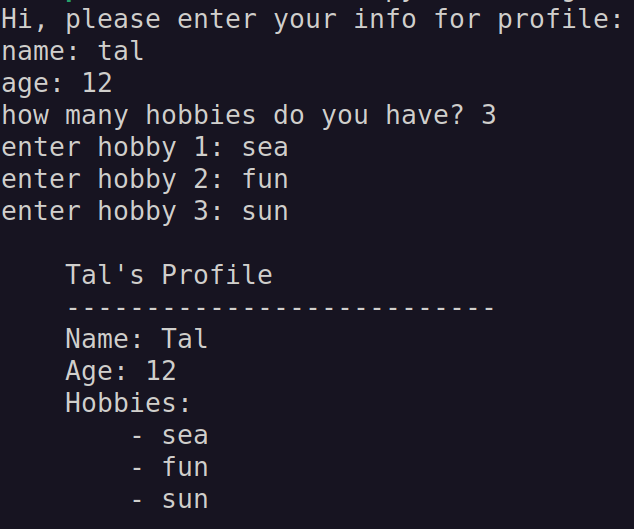
תרגיל

אם מספר התחביבים גדול מ 6, לקבוע ל 6.

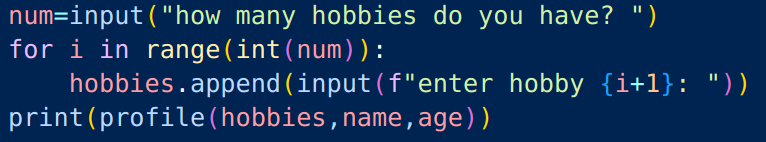
**תרגיל 11**

כשהתוכנית מבקשת את התחביבים, יש להציג את מספר האיטרציה.

דוגמת הרצה:



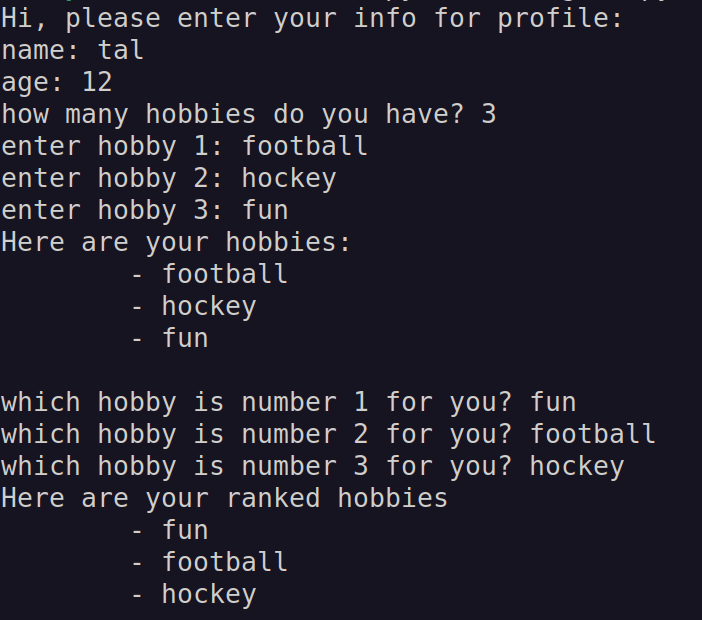
פתרון:



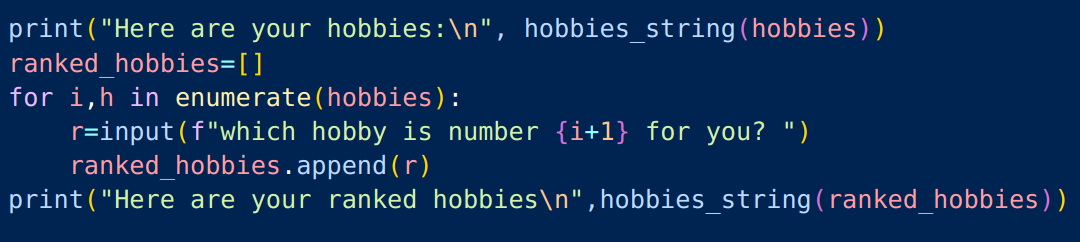
**תרגיל 13**

בקשו מהמשתמש את סדר העדיפות שלו וסדרו את הרשימה כך שהתחביבים יהיו בסדר העדיפות של המשתמש.

דוגמת הרצה:



פתרון:



**תרגיל 14:**

שפרו את הפונקציה שמפרמטת את רשימת התחביבים כך שתקבל כקלט מהמתכנת (לא מהמשתמש) את הפרמטרים הבאים ותפרמט בהתאם:

1. תווים לשים לפני כל תחביב (למשל – או \*)

2. פרמטר בוליאני לקבוע אם יהיה tab או לא

3. פרמטר בוליאני לקבוע עם יהיה מספור או לא

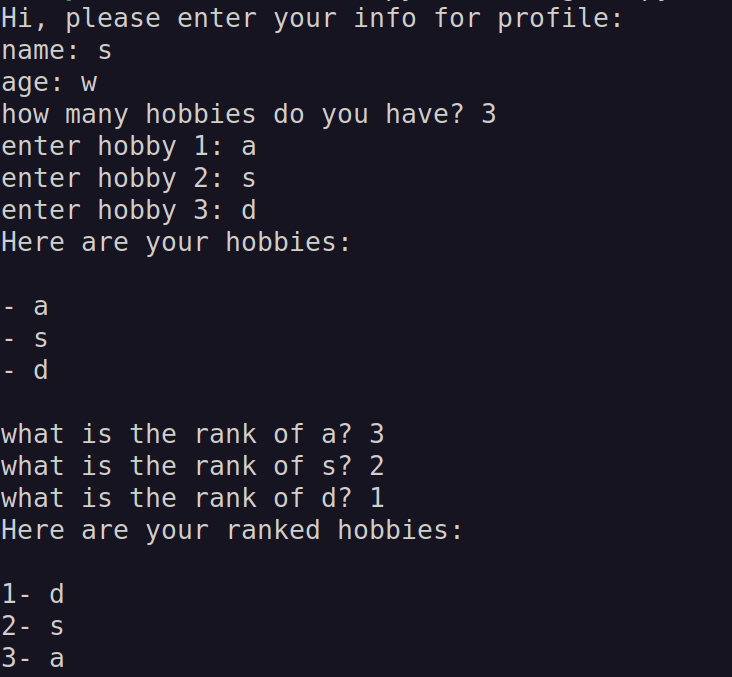
פתרון:



**תרגיל 15:**

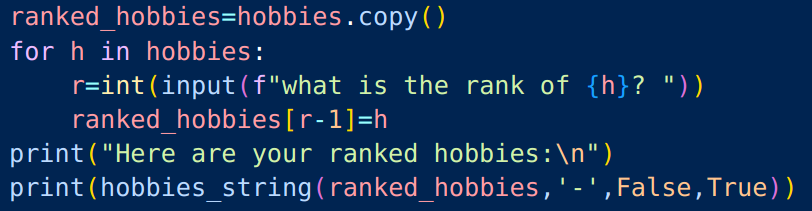
בקשו את הדירוג לכל תחביב.

דוגמת הרצה:



פתרון:

שימו לב שההשמה מתבצעת בעותק של הרשימה:



**תרגיל 16:**

כיתבו פונקציות לייצר נתונים פיקטיביים במקום לקבל אותם מהמשתמש:

פונקציה לייצור שם אקראי

פונקציה לייצור גיל אקראי

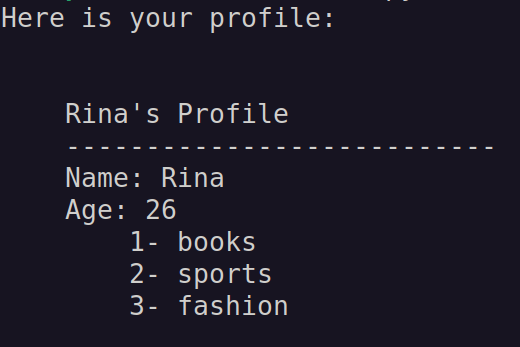
פונקציה לייצור תחביב אקראי

פונקציה לייצור רשימת תחביבים אקראיים

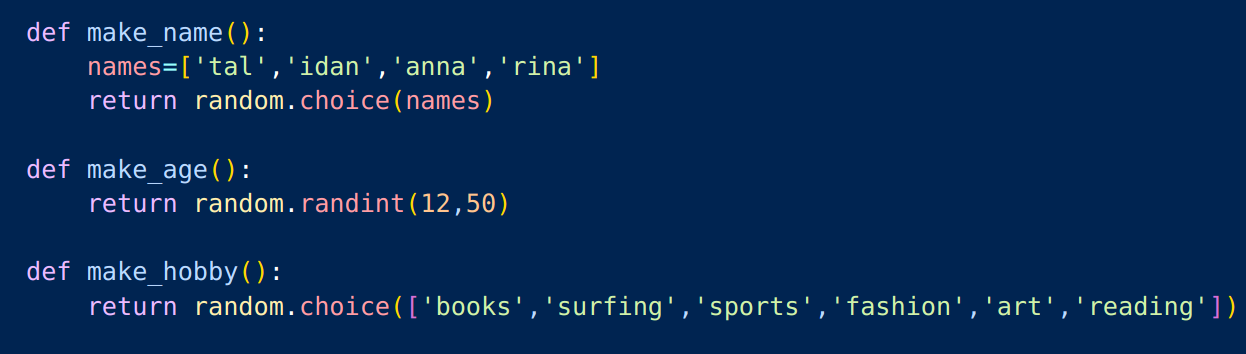
פונקציה לייצור רשימת תחביבים אקראיים מסודרת לפי העדפות אקראיות

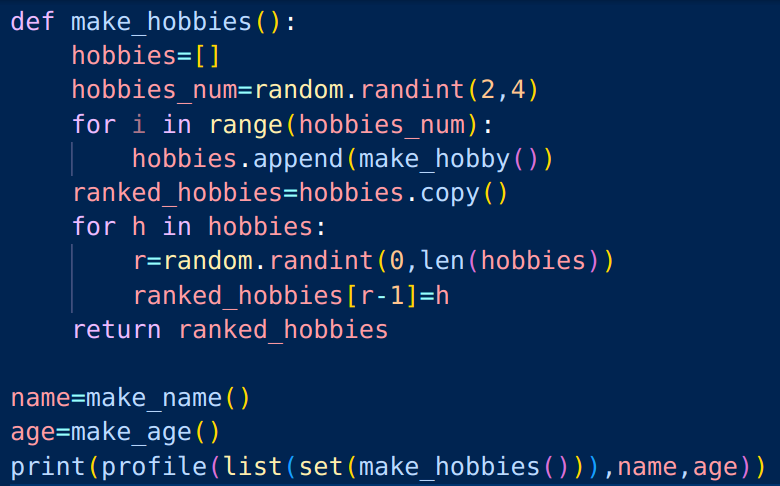
התוכנית צריכה לרוץ רק ע"י קריאה לפונקציות הנ"ל, והפונקציות שכתבתם בתרגילים הקודמים.

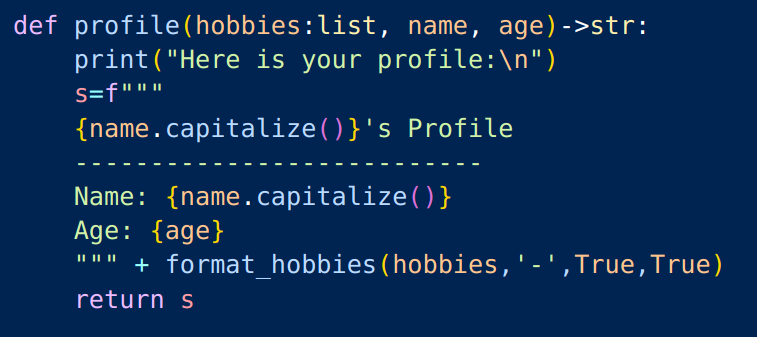
דוגמת הרצה:

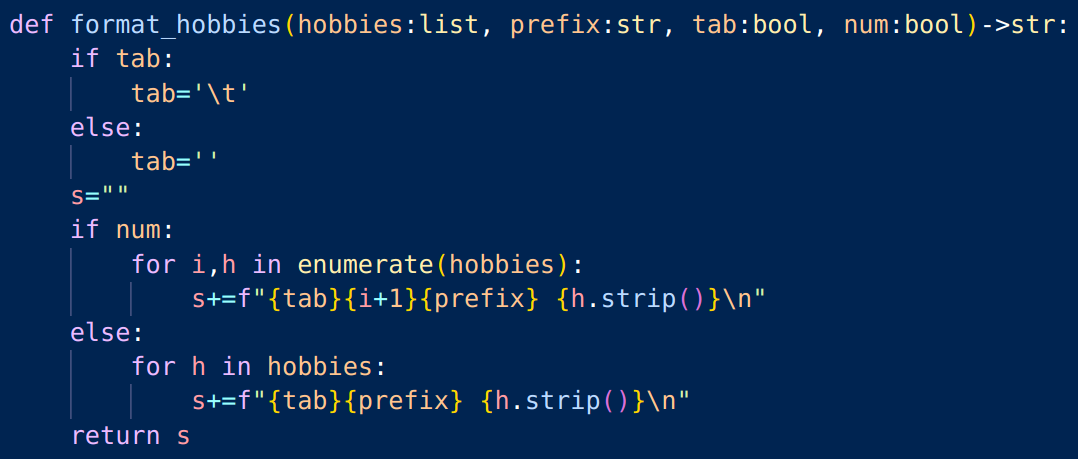


פתרון:









**תרגיל 17:**

נגדיר שפרופיל הוא מילון עם הנתונים הבאים: שם, גיל ורשימת תחביבים.

הגדירו פרופיל אחד.

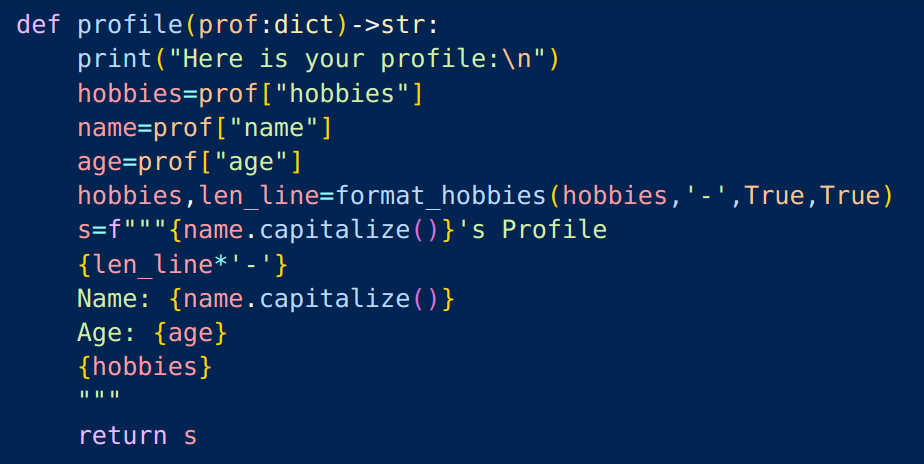
פתרון:



**תרגיל 18:**

שנו את הפונקציה profile כך שתקבל פרופיל (מילון) ותדפיס אותו.

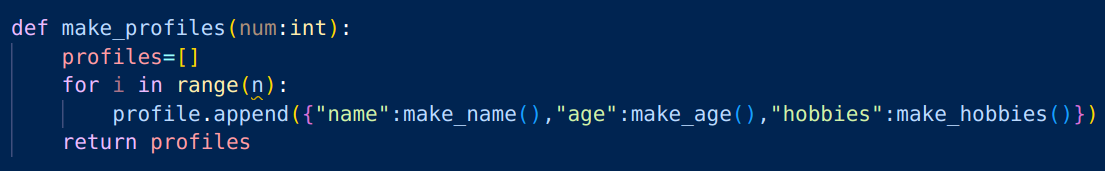
פתרון:



**תרגיל 19:**

כתבו פונקציה המחזירה מספר כלשהו של פרופילים.

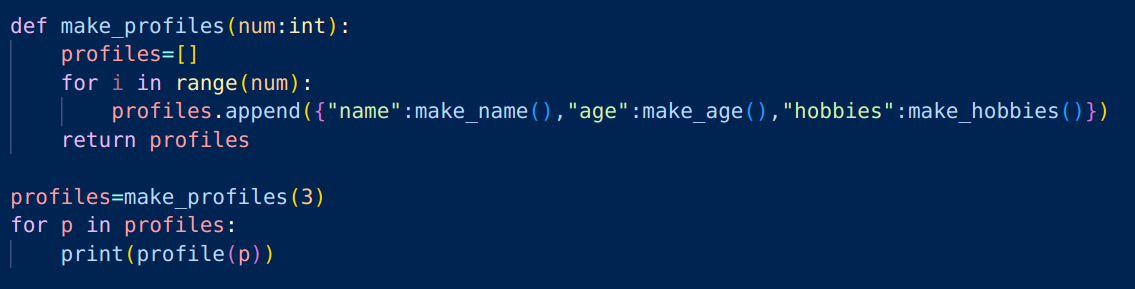
פתרון:



**תרגיל 20:**

השתמשו בפונקציה הנ"ל לייצר 3 פרופילים ולהדפיס אותם.

פתרון:



**תרגיל 21 (נפרד מהפרופילים)**

כתבו מילון שמייצג קלף. לקלף יש מספר בין 1-13, שם, וצורה (spade, heart, diamond, club).

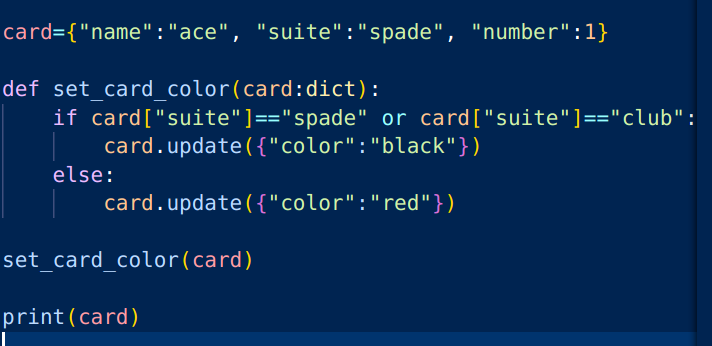
פתרון:



**תרגיל:**

אם קלף הוא spade או תלתן, הצבע שלו שחור, אחרת אדום. כתבו פונקציה המקבלת מילון קלף ומגדירה את הצבע.

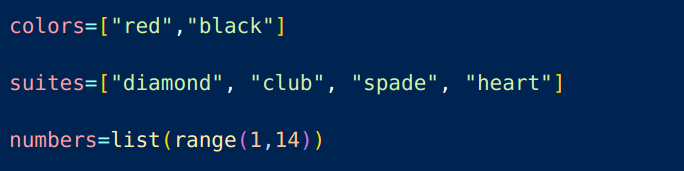
פתרון:



**תרגיל 22**

כתבו רשימות עם כל אחד מסוגי הנתונים: צבע מספר וצורה.

עבור המספרים, האם חייבים לכתוב ידנית את כולם?



**תרגיל 23**

מתוך 13 המספרים, 3 האחרונים נקראים בשם נסיך, מלך, מלכה, ואס הוא מספר 1. כיתבו מילון המתרגם מספרים אלה לשם המתאים.

פתרון:



**תרגיל 24**

האם אפשר לייצר רשימה של שמות מתוך המילון הנ"ל?

פתרון:



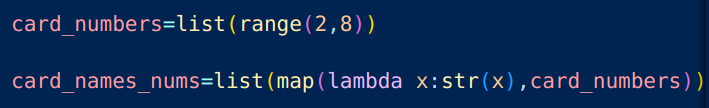
**תרגיל 25**

צרו רשימה של המספרים בין 2-10 אבל כמחרוזות, כלומר ["1", "2"...]



פתרון:





**תרגיל 26**

צרו מילון הממפה בין המספרים בין 2-10 ובין הרשימה הנ"ל.

פתרון:



**תרגיל 27:**

הוסיפו את ה items במילון בתרגיל הקודם למילון של שמות הקלפים.

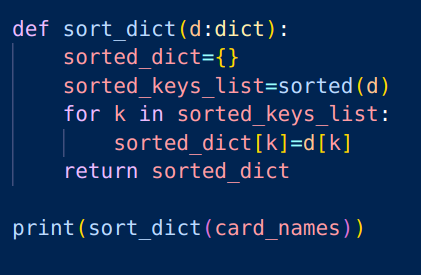
פתרון:



**תרגיל 28:**

שימו לב שבתרגיל הקודם, האס לא בסדר הנכון. כתבו פונקציה שממייינת את המילון.

פתרון:



**תרגיל 29:**

האם אפשר ששם המילון המסודר יהיה card\_games?

פתרון:



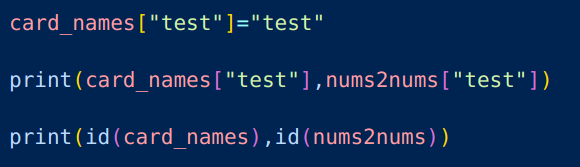
**תרגיל 30**

האם nums2nums ו card\_games הם אותו מילון, או עותקים נפרדים?

פתרון:

נוסיף פריט למילון אחד ונראה שגם המילון השני השתנה, כלומר הם שני שמות שונים לאותו מילון.

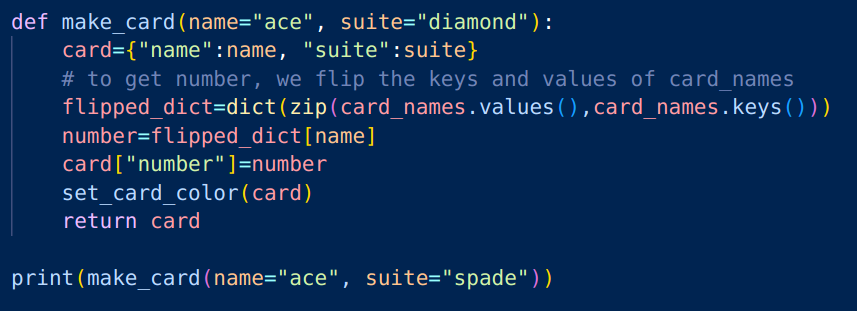
נבדוק שגם ה id שלהם, כלומר הכתובת בזיכרון של המילון, היא אותה כתובת.



**תרגיל 31**

צרו פונקציה לייצור קלף קלף עם שם, מספר, צבע וסוג הראשון מתוך המילון והרשימות שבניתם. הפונקציה תקבל שם וסוג ותחזיר מילון עם כל הנתונים.

פתרון:



**תרגיל 32**

צרו קלף עם שם, מספר וסוג הרביעי מתוך המילון והרשימות שבניתם.

**תרגיל 33:**

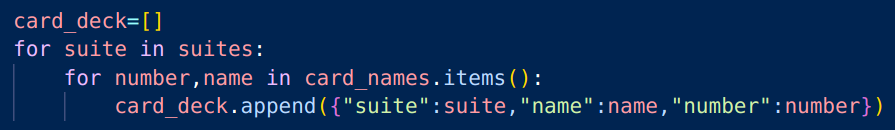
צרו קלף עם שם, מספר וסוג אקראי מתוך הפונקציה.

פתרון:



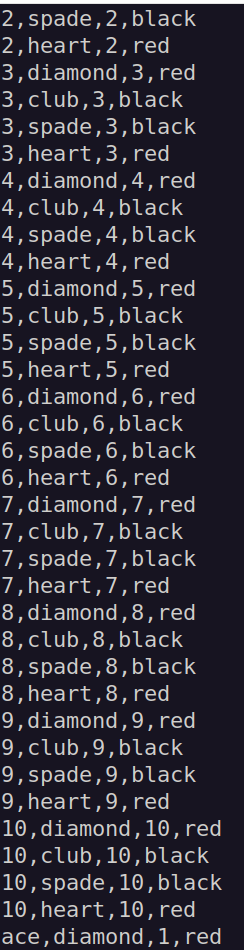
**תרגיל 34**

צרו חפיסת קלפים, 13 מכל סוג.

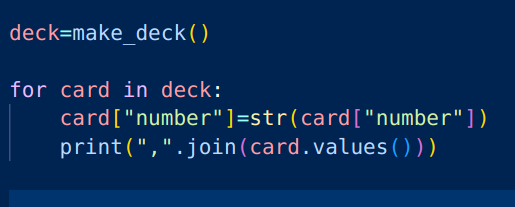


**תרגיל**

הציגו כל קלף בחבילה כ CSV

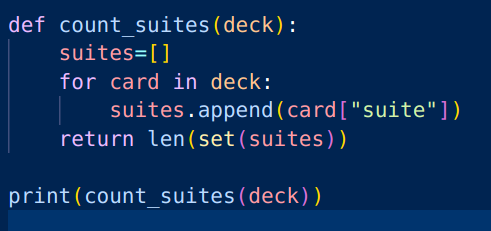


פתרון:

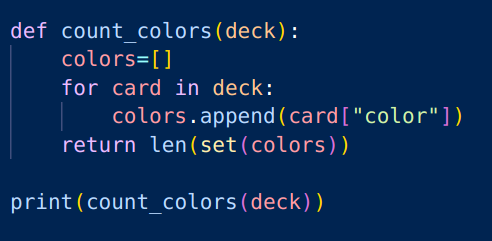


**תרגיל**

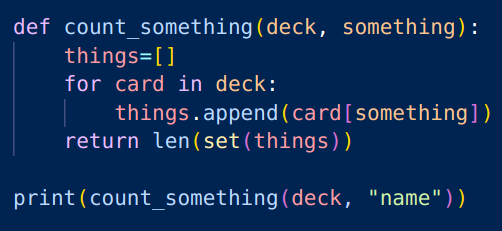
כמה סוגים שונים יש בחבילה?



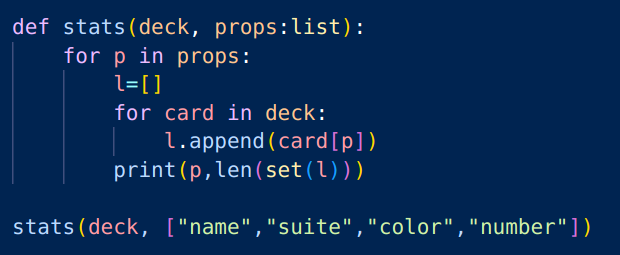
כמה צבעים שונים יש בחבילה?

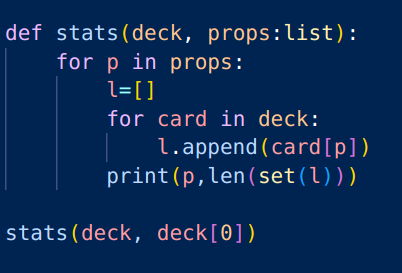


הכלילו את הפונקציה לכל פרמטר.

****

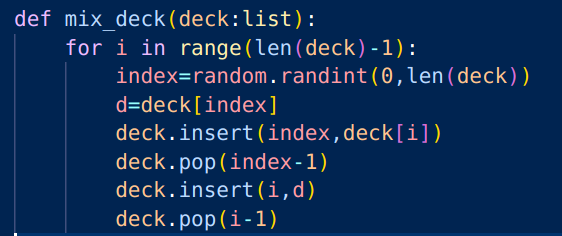
העבירו רשימה של תכונות וספרו אותם.





**תרגיל**

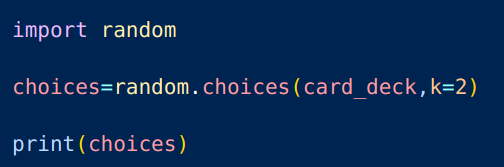
כתבו פונקציה "לערבב" את החבילה. בדקו.



**תרגיל 35**

בחרו 2 קלפים באופן אקראי מהרשימה והציגו אותם למשתמש.

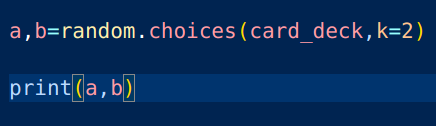
פתרון:



**תרגיל 36**

בחרו 2 קלפים באופן אקראי והכניסו אותם לשני משתנים.

פתרון:



**תרגיל 37:**

בחרו את שני המספרים של הקלפים האקראיים שבחרתם. הכניסו אותם למשתנים והדפיסו אותם.

**תרגיל 38:**

חשבו את הסכום של שני המספרים הנ"ל. אם הסכום גדול מ 21 רשמו "lose".

**תרגיל 39:**

אחרת, רשמו "do you want another card?"

**תרגיל 40:**

שנו את ההדפסה ל input וקבלו קלט. אם כן, בחרו עוד קלף והוסיפו אותו לסכום.

**תרגיל 41:**

הכניסו ללולאה שמתבצעת כמספר הקלפים בחפיסה את השלבים 38-40.

**תרגיל 42:**

הוסיפו תנאי עצירה אם סכום גדול מ 21 או אם סכום שווה ל 21 ואז תרשמו "win".

**תרגיל 43:**

הוסיפו בחירה של עוד 2 קלפים לפני הלולאה, וקלף נוסף אם בוחר המשתמש. הוסיפו סכום נוסף והשוו ל 21.

מתקדם:

הוסיפו תכונה של "חבר" לפרופיל. חבר הוא מי שיש לו לפחות תחביב משותף אחת.

אחר כך חברים לפי דירוג כמה תחביבים משותפים.

גרף

בחירת תחביב לפי תחביבים פופולרים (משותפים)

שפרו את הפורמט של הפרופיל כך שהקו מתחת לכותרת יהיה בדיוק ברוחב של הטקסט.

בצעו בדיקת איות על התחביבים שמכניס המשתמש (יש להשתמש בספריה חיצונית).

אם יש הבדל בין תוצאת הבדיקה לקלט, יש להציע תיקון ולתת למשתמש לבחור אם לתקן או לא ("did you mean...")

הרשימה תכיל רק מילים שתוקנו.