גרסה 0.10 – 22/10/2010

P y t h o n

ש ו ש ן כ ה ן

מסמך זה הורד מהאתר www.underwar.co.il

אין להפיץ מסמך זה במדיה כלשהי, ללא אישור מפורש מאת המחבר.

מחבר המסמך איננו אחראי לכל נזק, ישיר או עקיף, שיגרם עקב השימוש במידע המופיע במסמך, וכן לנכונות התוכן של הנושאים המופיעים במסמך. עם זאת, המחבר עשה את מירב המאמצים כדי לספק את המידע המדויק והמלא ביותר.

כל הזכויות שמורות ל**שושן כהן**

מסמך זה מכיל הסברים אודות Python. לא נדרש ידע קודם שידוע לי עליו להבנת המסמך, במידה ואכן קיים כזה אשמח אם תוכלו לשלוח לי בקשות להסבר בפורום של UnderWarrior – http://www.udnerwar.co.il/forum/ או במייל (psclil at gmial dot com).

1 הקדמה – שפת python

Python – תיאור ושימושים

שפת python הינה שפה שנתמכת במערכות רבות ומגוונות ומשמשת לצרכים רבים, בניית אתרים ושרתים, כריית נתונים, יישום אלגוריתמים ובניית סקריפטים הם רק אחדים מהם.

השפה הינה פתוחה (קוד המקור שלה ושל החבילות הרשמיות שלה הינו פתוח לכולם, זוהי שפת Open Source) ומכילה הפצות רשמיות עבור Windows, מקינטוש והפצות רבות מאוד של Linux (כולל למשל הפצות עבור אנדרואיד וטלפונים חכמים רבים מבוססי לינוקס!) – *Python* מאפשרת (אך לא מכריחה) למתכנת בשפה לכתוב פעם אחת ולהשתמש על מערכות רבות – הספרייה הרשמית שלה נתמכת בהפצות הרשמיות וקוד פשוט אמור לעבוד.

לשפה שתי גרסאות שמתוחזקות וממשיכות להתפתח בעת כתיבת המסמך: Python3 (הגרסה אחרונה הינה 3.2) ו-Python2 (הגרסה האחרונה הינה 2.7). מסמך זה ייתן את דוגמאות הקוד שבו עבור גרסה 2.7 מאחר והיא תואמת לאחור ובשימוש הרבה יותר נרחב בעת כתיבת המסמך (וזאת מאחר ובהרבה מאוד חבילות שתואמות לגרסאות 2.3, 2.4 ו-2.5 הספיקו להתפרסם).

Python היא שפה הן לפיתוח מהיר ויעיל והן לפיתוח נכון מתודולוגית וארוך טווח

Python הינה שפה אשר אינה דורשת הידור, כלומר ניתן פשוט לכתוב בה קוד ומבלי לדאוג להדר אותו (להמיר קובץ קוד לקוד שהמחשב בינארי שמערכת ההפעלה והמעבד מבינים) ולקשר אותו (לחבר מספר קבצים מקומפלים לקובץ אחד אשר ניתן להרצה). יש לכך יתרונות וחסרונות ועליהם ראו בפרק הבא.

אחד העקרונות החשובים בעת פיתוח בשפה הינו Code Reuse, כלומר שימוש חוזר בקוד, שימוש בספריות שכבר נכתבו. היתרון בעקרון זה הוא עצום בקרב מתכנתים מנוסים ומאפשר פרויקטים קטנים שעושים המון. דוגמא אחת מבין המון דוגמאות: אין צורך לכתוב אלגוריתם שמוציא ציון Page Rank של גוגל, יש כבר דוגמאות קוד ואף ספריות שעושות זאת! אין צורך לממש בעצמנו שרת אינטרנט, יש ספריה שמבצעת זאת.

Python היא גם שפה מונחית עצמים, כלומר *אובייקטים מורכבים* (כגון "שרת אינטרנט") ניתן להרחיב, לשנות ולהוסיף להם פונקציונאליות – ויחד עם זאת לאפשר להשתמש בהם לפי אותו *חוזה שימוש* שהוגדר להם מהתחלה - עובדה זו מאפשרת לנו להחליף חלקים התנהגויות וחלקים קטנים בקוד קיים

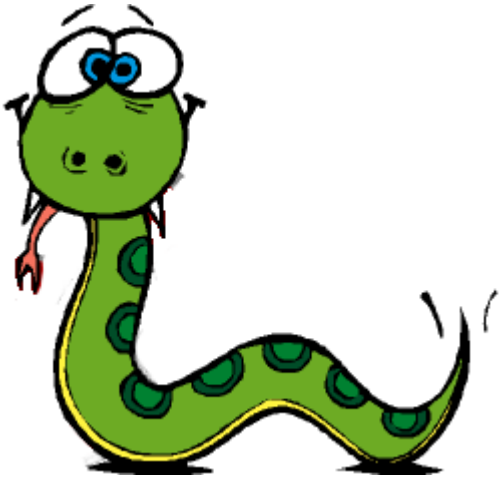
Python על Windows

כל החבילות הרשמיות של פייטון נתמכות בווינדוס, וישנן אף חבילות רשמיות אשר מיועדות למערכת ווינדוס בלבד כגון *winreg* אשר מיועדת לקריאה ולכתיבה מה-Registry של המערכת.

חשוב לציין שבכתיבה ב-Python אנו משתמשים רבות בחבילות חיצוניות, למשל חבילה ליצירת מסמכי PDF – אך פעמים רבות אנו מגלים כי מפתחי החבילות הסתמכו על ספריות חיצוניות לשפה או על תוכניות שאינן קיימות עבור Windows – ולכן אין הפצה לספריה עבור ווינדוס, או שישנה אחת אבל לא מצוין בה שאינה עובדת על ווינדוס.

הדרכים להתמודד עם מקרים מהסוג הזה הינן:

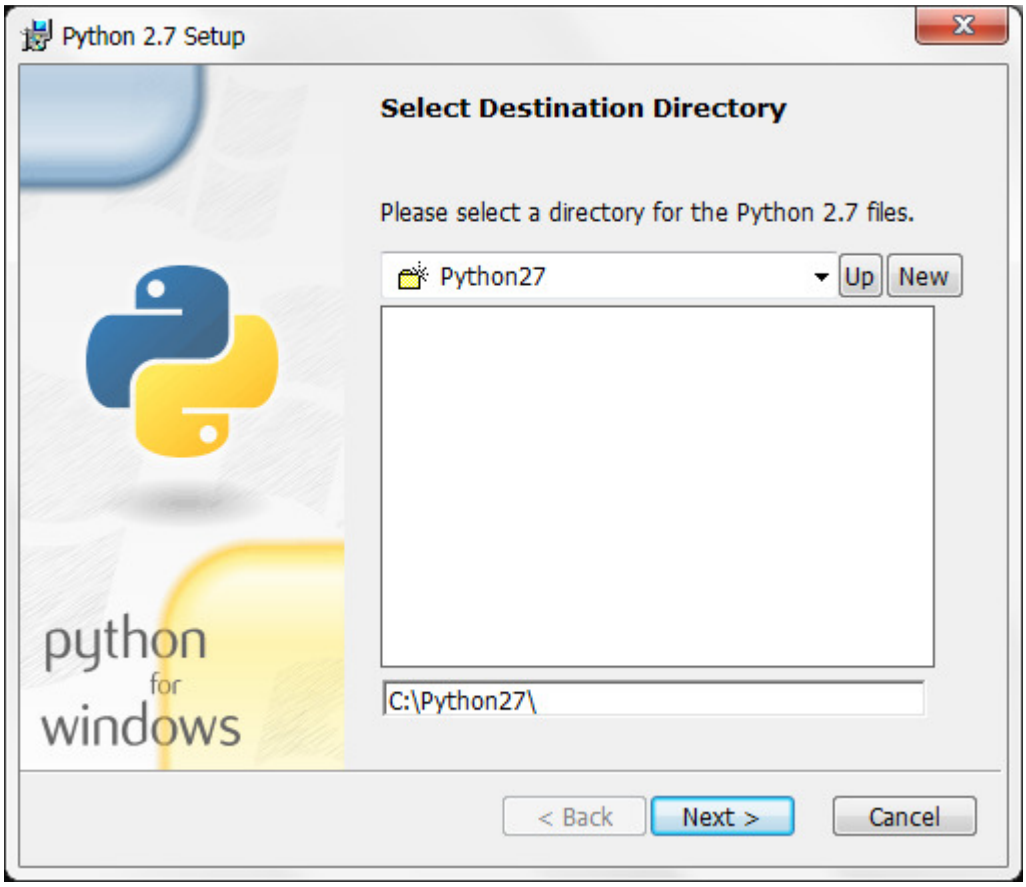
1. לחפש התפצלות של הספרייה המקורית עבור מערכת ההפעלה הנדרשת (בלבד או גם).
2. להבין אם הבעיה היא רק חוסר בדיקה והתאמות קטנות למערכת ההפעלה הנדרשת – ובמידה וכן לבצע אותם.
3. לחפש ספריה ופתרון חלופי, לרוב אף תגלו שלא מעט אנשים היו במצב שלכם לפניכם!



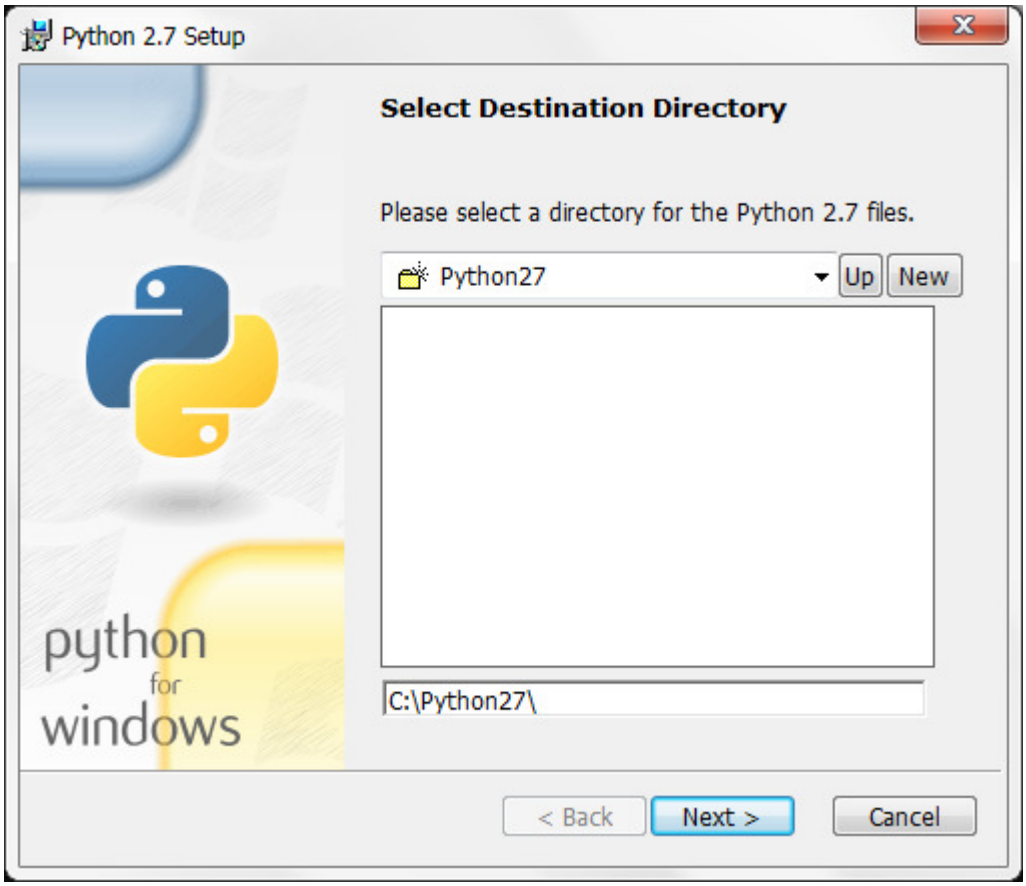
Python על מכשירים חכמים

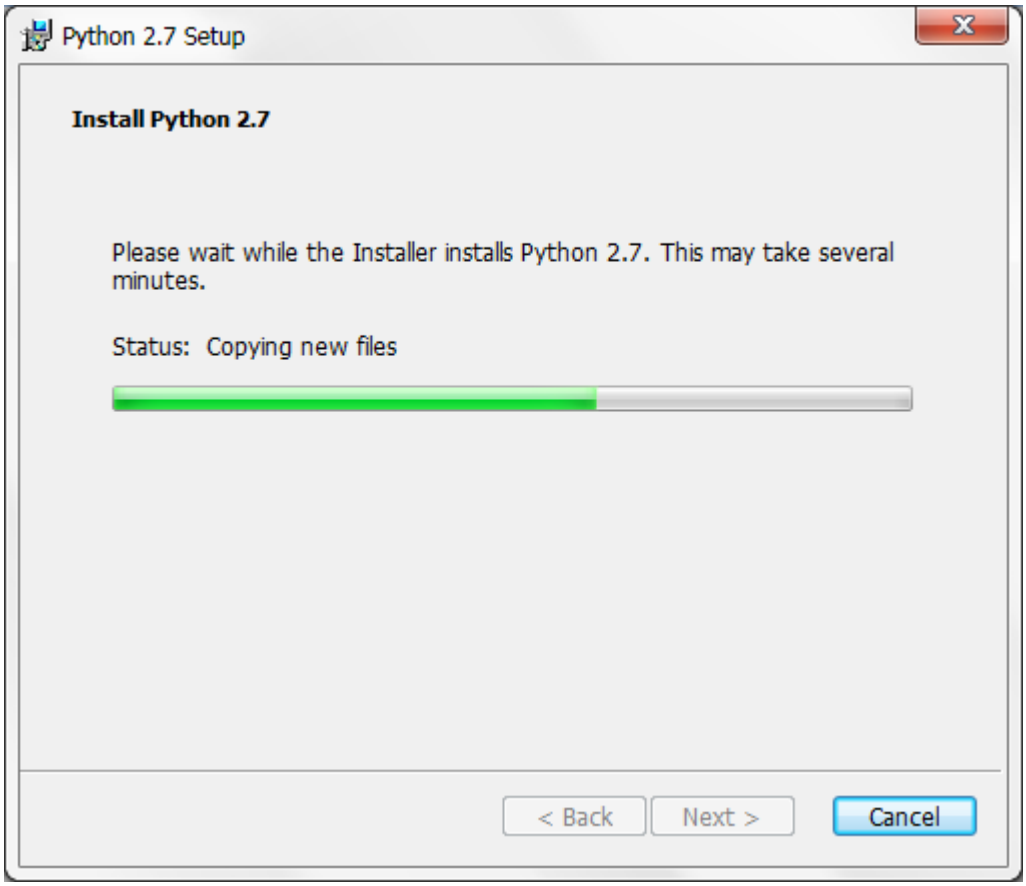
חשוב לציין, אמנם יש הפצות עבור מכשירים חכמים כגון אנדרואיד, אך אין בהפצות אלו כל יכולת מיוחדת לטלפון (אשר נתמכות באופן רשמי בחבילות הרשמיות של השפה), ועבור סוגים של טלפונים יש להשתמש בספריות חיצוניות לביצוע פעולות שונות אשר לא יפעלו בגלל שהמימוש שלהן שונה עבור מערכת ההפעלה של המכשיר החכם. מעבר לכך כמובן שעבור תכונות שהן רק של הטלפון (למשל רטט של המכשיר, צלצול וכו') – יהיה צורך בספריות נוספות – במקרים מסוימים בהחלט אפשרי שלא יהיו כאלו אשר יהיו במצב מספיק מתקדם לשימוש.

2 התקנת Python



תהליך ההתקנה הינו פשוט למדי, יש פשוט ללחוץ על Next בכל שלב (במידה ואין לכם הרשאות מנהל יש לבחור להתקין עבור המשתמש הנוכחי בלבד).

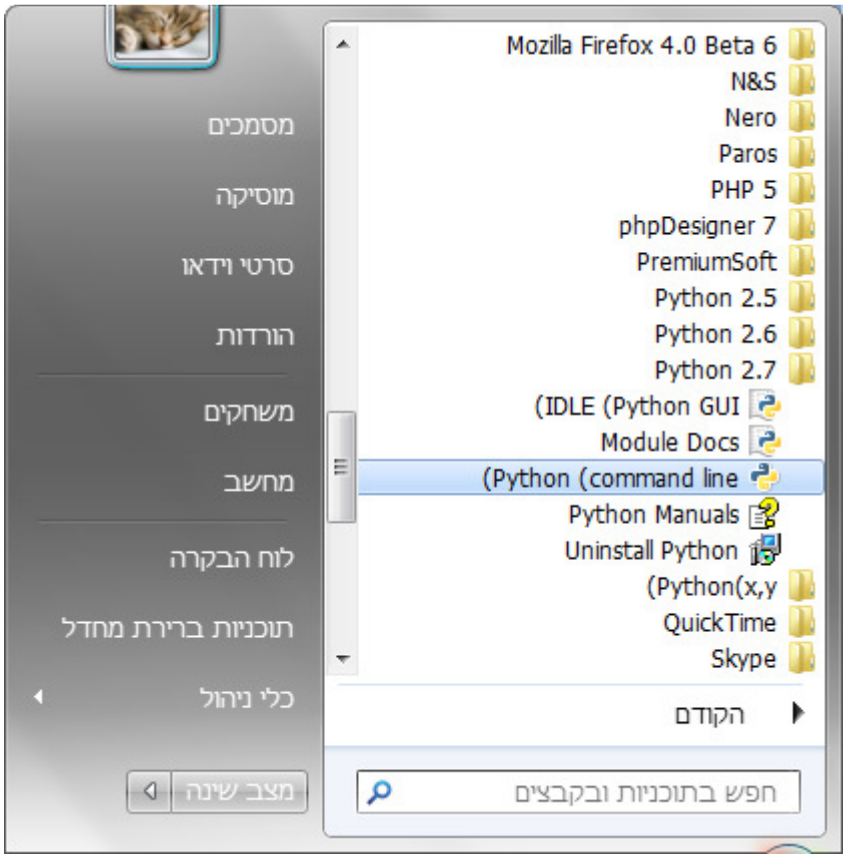

Python 2.7 Setup
Select Destination Directory
Please select a directory for the Python 2.7 files.
Python27
Up
New
C:\Python27\
python
for
windows
< Back
Next >
Cancel


Python 2.7 Setup
Install Python 2.7
Please wait while the Installer installs Python 2.7. This may take several minutes.
Status: Copying new files
< Back
Next >
Cancel

Python 2.7 Setup
Completing the Python 2.7 Installer
Special Windows thanks to:
Mark Hammond, without whose years of freely shared Windows expertise, Python for Windows would still be Python for DOS.
python
for
windows
Click the Finish button to exit the Installer.
< Back
Finish
Cancel

3 Python Shell – שורת פקודה אינטראקטיבית

לאחר שהתקנתם את Python במערכת – הפעילה את כלי שורת הפקודה:

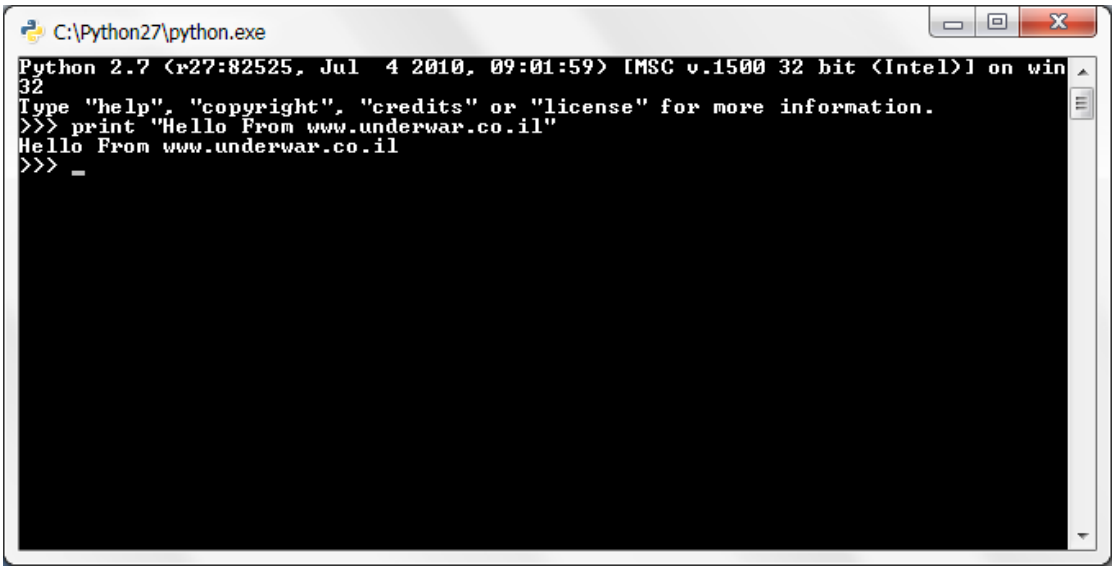


הקלידו את הפקודה (אין צורך להקליד את החצים בהתחלה!):

```
>>> print "Hello from www.underwar.co.il"
```

ותקבלו את הפלט

```
>>> print "Hello from www.underwar.co.il"
Hello From www.underwar.co.il
>>>
```



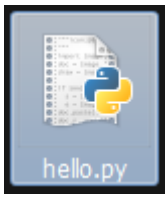
וההסבר הפשוט הוא – הפקודה print מקבלת רשימת אובייקטים (ערכים או משתנים, עליהם נלמד בהמשך) – ומדפיסה אותם לפלט של התוכנית, בסוף הפלט הפקודה יורדת שורה, במקרה של שימוש בכלי האינטראקטיבי של Python, אך לא תמיד בתוכניות אמיתיות שנכתוב הפלט של התוכנית פשוט מאוד הולך למסך.

שלושה החצים שבתחילת השורה הבאה לא נוספו בגלל הפקודה אלא בגלל שאנו במצב אינטראקטיבי והתוכנית רוצה "להראות" לנו שניתן כבר להקליד את הפקודה הבאה.

הערה – הפעלת Python

במקרה זה אמנם בחרנו להפעיל את Python על ידי תפריט ההתחלה אך חשוב לציין שישנן עוד דרכים להפעיל את התוכנית:

1. דרך שורת הפקודה – בהרצת הפקודה python (או python.exe) – במידה והגרסה הרצויה היא היחידה שמותקנת במחשב – או c:\python27\python.exe אם ישנן גרסאות נוספות מותקנות אך ברצונכם להפעיל את גרסה 2.7 (התיקייה היא כמובן תיקיית ההתקנה שבחרתם בפרק הקודם!).
2. יצירת קובץ Python – והרצתו (קובץ Python הינו כל קובץ המסתיים ב-py., לדוגמא צרו קובץ בעזרת פנקס רשימות בשם hello.py, שימו בו את הפקודה שלמדתם קודם לכן, הפעילו אותו – ותראו [כנראה לפרק זמן קצרצר – הסבר על כך בהמשך] חלון שחור קובץ ובו התוכן Hello From www.underwar.co.il)



הערה – מחרוזת

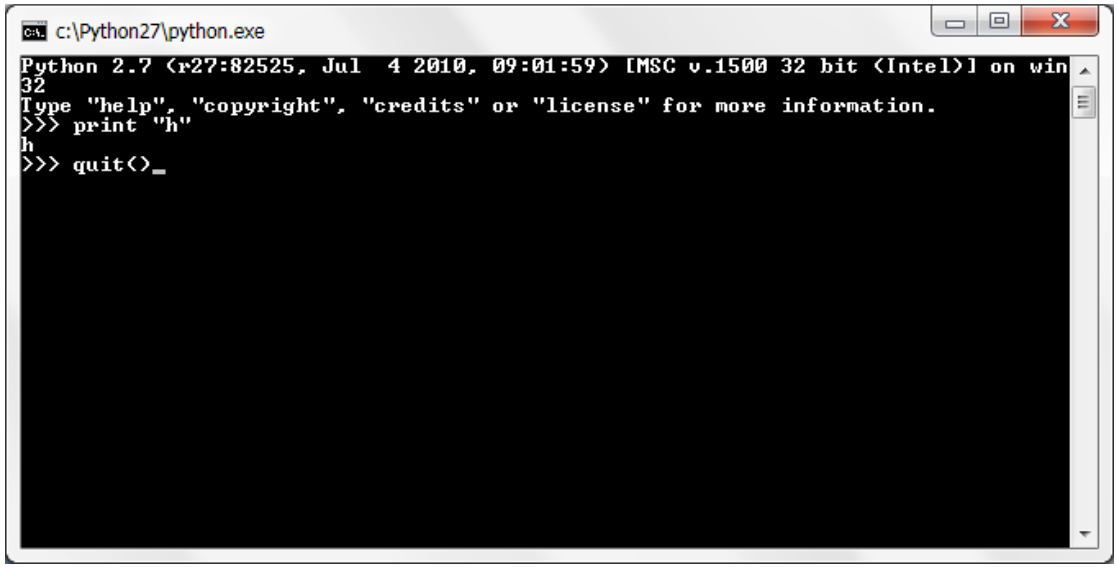
מחרוזת הינה פשוט מאוד טקסט – בפקודה לדוגמא שהבאנו רצינו להדפיס את הטקסט Hello From www.underwar.co.il למסך ולצורך כן העברנו אותו לפקודה

print, אך שמנו אותו בין גרשיים: "Hello From www.underwar.co.il" – זאת הדרך לומר בשפת Python כי מדובר בערך מסוג מחרוזת, עוד על מחרוזות ועל סוגי משתנים בפרק הבא.

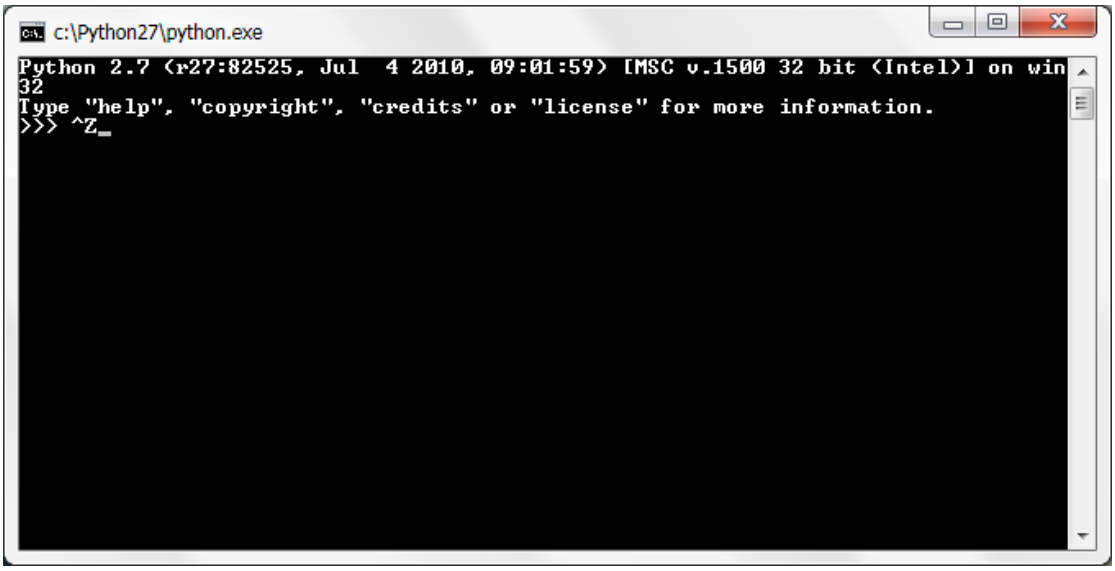
סיום התוכנית

ישנן שתי דרכים לסיים את התוכנית בשפת python.

הראשונה היא הפונקציה quit:



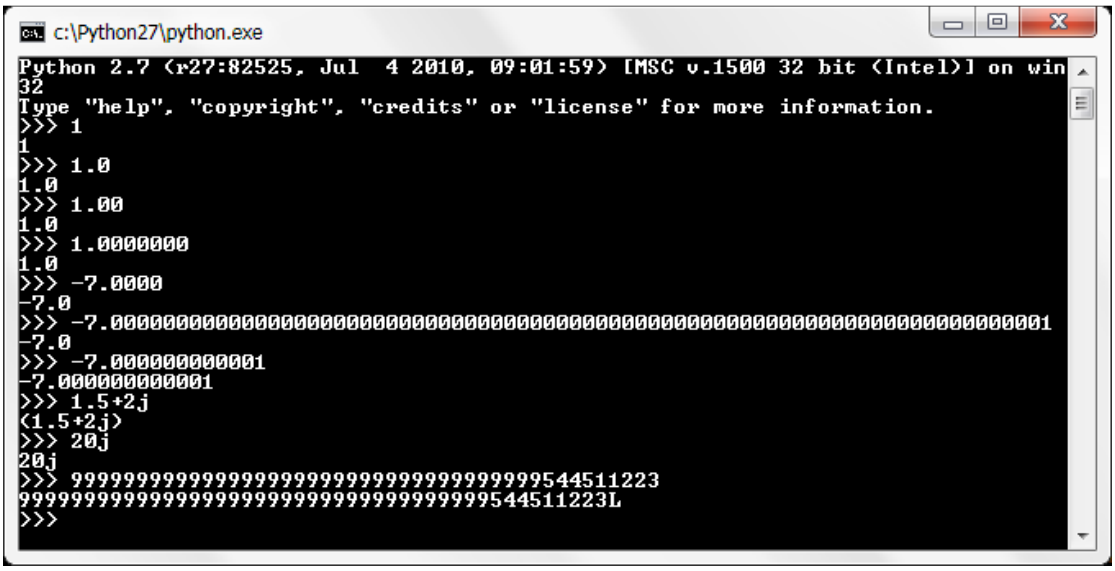
הדרך השנייה והיפה יותר היא סיום הקלט לתוכנית – במידה ומסמנים לתוכנית את "סוף הקלט" – שהוא שווה ערך לסוף הקובץ במידה ומריצים סקריפט השמור בקובץ py. הדרך לסמן זאת למפרש של Python היא שילוב המקשים Ctrl+Z – אשר יוצרת סימן מיוחד שפירושו סוף הקובץ (EOF).


c:\Python27\python.exe
Python 2.7 (r27:82525, Jul 4 2010, 09:01:59) [MSC v.1500 32 bit (Intel)] on win
32
Type "help", "copyright", "credits" or "license" for more information.
>>> ^Z

4 סוגי ערכים

4.1 מספרים ב-Python

מספרים ב-Python ניתנים לייצוג פשוט על ידי ספרות (0 עד 9), סימן שלילה (-) ונקודות עשרוניות (.) והאות j לאחר מספר, שפירושה היא שהמספר הוא מספר מרוכב (אין חובה להכיר עבור הבנת המסמך, 2j הוא השורש הריבועי של מינוס 4). לדוגמא: 2, 4, 81122, -72.64, 10j

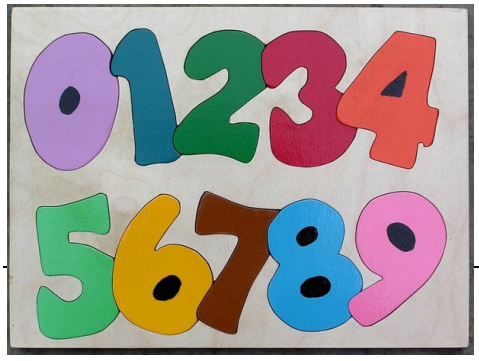


סוגי מספרים:

1. int – מספר שלם
2. long – מספר שלם גדול
3. float – מספר עשרוני
4. complex – מספר מרוכב

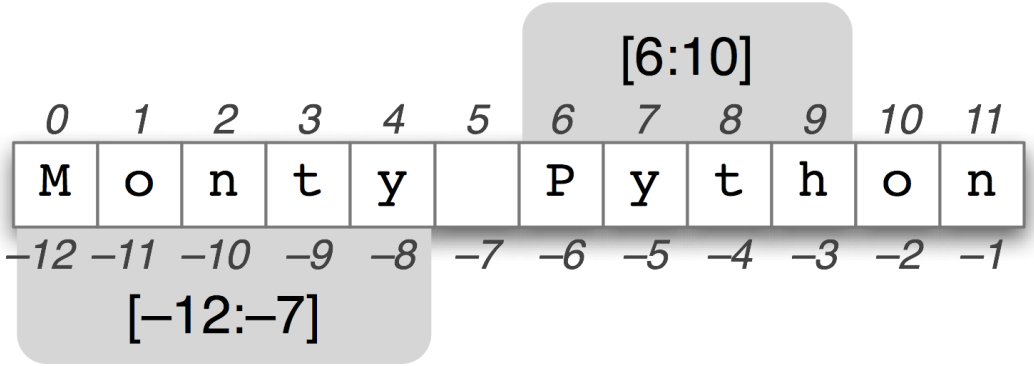
הערות:

- שימו לב לדוגמא האחרונה – Python בתור ברירת מחדל, בניגוד לשפות רבות אחרות, מסוגלת לייצג מספרים מאוד (מאוד, מאוד, מאוד) גדולים – אם לדייק הם מוגבלים רק בכמות הזיכרון של התוכנית (לא מוגבלים לכאורה) – אך יש פונקציות שמוגבלות לעיתים בגודל המספר שהן יכולות לקבל בגלל המימוש שלהן.
- (אין צורך בשפה להבדיל בין int ל-long, הם פשוט מיוצגים בצורה מעט שונה, אך ערכים יומרו לנו אוטומטית במידת הצורך)



- מספרים מרוכבים מיוצגים באות j, לא באות i.

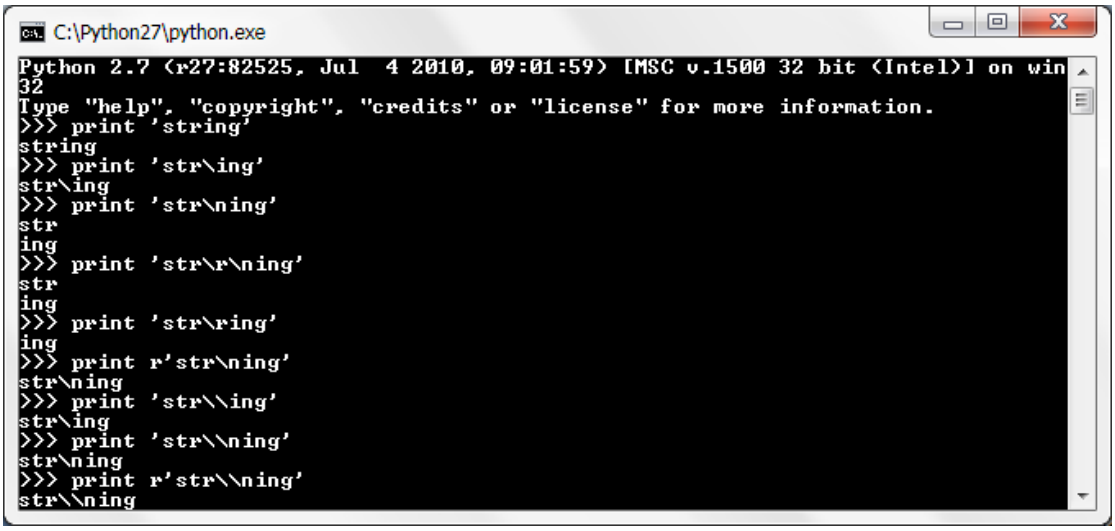
4.2 מחרוזת ב-Python - string



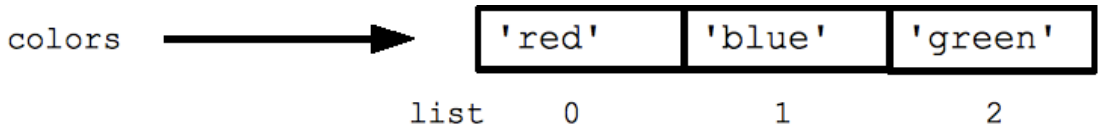
מחרוזות הן פשוט רשימת אותיות, ב-Python ישנם שני סוגי מחרוזות: מחרוזות רגילות, ומחרוזות Unicode. מחרוזות Unicode יכולות להכיל גם תווים מעבר לקבוצת ASCII. כלומר תווים בעברית, ערבית, סינית כמובן אנגלית, וכו'. לעומתן מחרוזות רגילות מסוגלות להכיל אנגלית בלבד.

שפת פייטון מאפשרת להמיר בין קידודים שונים בצורה פשוטה כולל בדיקת שגיאות, ופתרון אוטומטי שלהן.

- הסימון של מחרוזת הוא על ידי גרש בודד העוטף אותה ('string' מייצג את המחרוזת string) או על ידי הסימן גרשיים (")
- הסימון של מחרוזת Unicode הוא פשוט מאוד על ידי הסימן u בתחילת המחרוזת: u'unicode'.
- ניתן לבצע escaping על ידי סימן הסלאש (/ - כמו בשפות רבות אחרות) – כך למשל ניתן לשים גרש בתוך מחרוזת שמסומנת על ידי גרש. בעזרת אותו סימן ניתן להגדיר תווים מיוחדים, לדוגמא: \n מסמן ירידת שורה, ומכיוון שלסלאש משמעות מיוחדת יש צורך לבצע escaping על מנת לייצג אותו, לדוגמא: \\ משמעותו \ בודד.
- ניתן להגדיר לשפה להתעלם מסימנית מיוחדים ובכך לבטל את הצורך בשני סלאשים על מנת לייצג סלאש כך: r'\n\\\asdas\as' – ישנו r לפני המחרוזת ולכן הערך שלה הוא \n\\\asdas\as.
- ניתן להגדיר לשפה מחרוזת שלא מסתיימת בירידת השורה אלא פשוט ממשיכה על ידי סימון משולש של ההתחלה והסוף שלה (" " " או ' ' ' – ללא הרווחים כמובן) – בצורה כזו נוח לכתוב מחרוזות תיעוד עבור פונקציות ומחלקות בשפה, או לשמור במשתנים ערכים קבועים מרובי שורה מבלי לבצע עבורם escaping לירידות השורה.



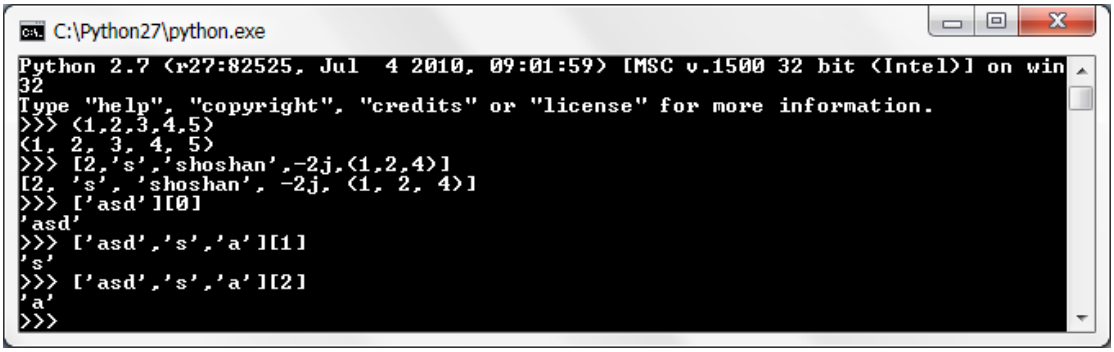
4.3 רשימות וטופלים ב-Python – lists and tuples



רשימות וטופלים (lists and tuples) ב-Python הן פשוט קבוצות ערכים בסדר קבוע, כאשר ההבדל בין רשימה לטופל הוא שרשימה ניתנת לשינוי (mutable) וטופל הוא קבוע (immutable), כלומר לא ניתן להוסיף לו ערכים, להסיר ממנו ערכים או לשנות ערכים בו.

רשימות ב-Python יכולות להכיל כל ערך וניתן לערב בהן מספר סוגי ערכים.

לדוגמא:



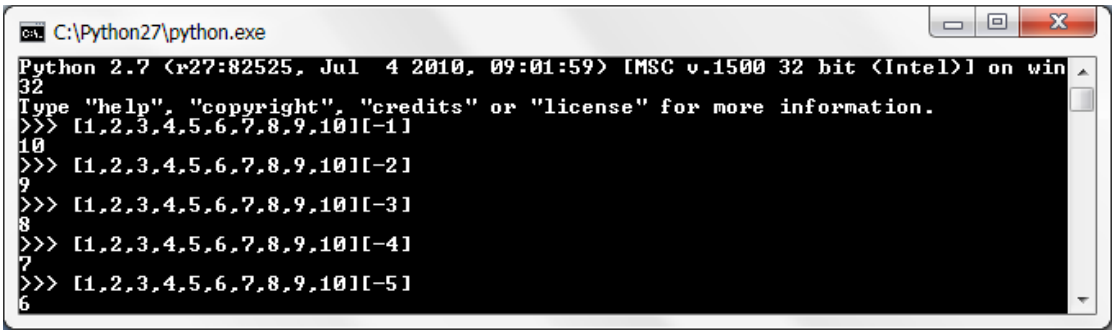
שימו לב שאין בעיה לשים בתוך רשימה אחת גם מחרוזת, גם מספר וגם רשימות נוספות וזאת בניגוד למערכים בשפות שונות בהן ניתן לשים רק ערך מסוג אחד במערך. בשפות אחרון גם במידה וכבר נוצרה רשימה בעלת מספר סוגי ערכים יש

צורך להגדיר למהדר מאיזה סוג ערך ברגע שמתייחסים אליו – אך זהו לא המצב בפייטון (ולא רק בגלל שבפייטון יש מפרש ולא מהדר).

כלומר שפת פייטון שומרת עבורנו מה הוא הסוג של כל ערך ברשימה.

קבלת ערך בודד מתוך הרשימה מתבצעת על ידי האופרטור [] כאשר הפרמטר בין הסוגריים המרובעים הוא מספר שלם, כפי שניתן לראות בדוגמא האינדקס שמייצג את האיבר הראשון ברשימה הוא 0, האיבר הרביעי נמצא באינדקס 3 וכך הלאה.

בנוסף פייטון נותנת לנו גמישות גדולה מאוד בכך שניתן לגשת לערכים בהתאם לגודל הרשימה מבלי לחשב אותו. לדוגמא ניתן לגשת לערך האחרון על ידי האינדקס 1-, אחד לפניו על ידי 2- וכו'.



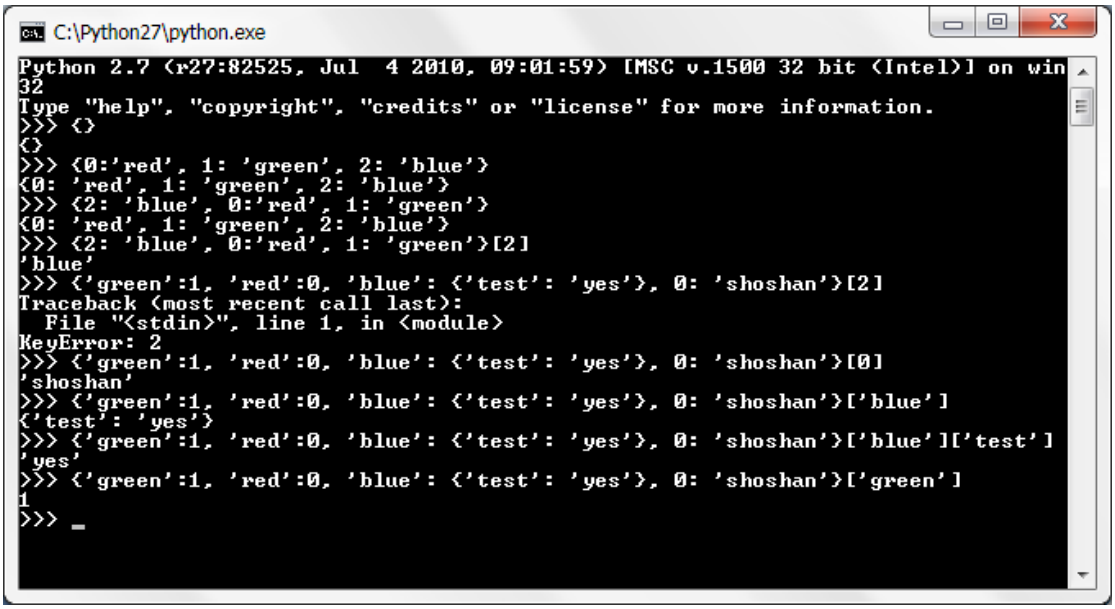
4.4 מילונים בשפת Python

מילונים הם דרך נוחה ויעילה לשמור ערכים לפי מתפתחות בצורה כזו שניתן יהיה לגשת אליהם במהירות וביעילות. בנוסף בצורה כזו לא ניתן לקבוע ערך כפול לאותו מפתח ולא ניתן לקבוע לשפה באיזה סדר לשמור את הערכים במילון.

המילון הוא מבנה נתונים מפורסם הידוע בשם Dictionary, HashTable וכו'.

הגדרת מילות מתבצעת על ידי הסימנים { } והגישה לערך הינה על ידי סוגריים מרובעים כמו ברשימות ([]) כאשר הפרמטר שמהווה את האינדקס הינו המפתח.

גישה לאינדקס שלא קיים תגרום לשגיאה – בהמשך המסמך יוסבר כיד ניתן להימנע ממנה על ידי בדיקה מוקדמת.


C:\Python27\python.exe
Python 2.7 (r27:82525, Jul 4 2010, 09:01:59) [MSC v.1500 32 bit (Intel)] on win
32
Type "help", "copyright", "credits" or "license" for more information.
>>> {}
{}
>>> {0:'red', 1: 'green', 2: 'blue'}
{0: 'red', 1: 'green', 2: 'blue'}
>>> {2: 'blue', 0:'red', 1: 'green'}
{0: 'red', 1: 'green', 2: 'blue'}
>>> {2: 'blue', 0:'red', 1: 'green'}[2]
'blue'
>>> {'green':1, 'red':0, 'blue': {'test': 'yes'}, 0: 'shoshan'}[2]
Traceback (most recent call last):
File "<stdin>", line 1, in <module>
KeyError: 2
>>> {'green':1, 'red':0, 'blue': {'test': 'yes'}, 0: 'shoshan'}[0]
'shoshan'
>>> {'green':1, 'red':0, 'blue': {'test': 'yes'}, 0: 'shoshan'}['blue']
{'test': 'yes'}
>>> {'green':1, 'red':0, 'blue': {'test': 'yes'}, 0: 'shoshan'}['blue']['test']
'yes'
>>> {'green':1, 'red':0, 'blue': {'test': 'yes'}, 0: 'shoshan'}['green']
1
>>> _

4.5 המשך יבוא

המשך יבוא