# pandas

פנדס היא ספריית קוד פתוח מהירה, עוצמתית, גמישה ופשוטה לשימוש עבור ניתוחים ומניפולציות על נתונים, והספרייה בנויה על בסיס השפה פייתון.  
הספרייה שימושית מאוד בהרבה תחומים, בניהם: כלכלה , חיזוי מניות, מדעי המוח, סטטיסטיקה, פרסום, ביג דאטה ועוד.   
היות והספרייה ענקית אנחנו נתמקד על הפונקציות העיקריות בה, אם תרצו לקבל עוד מידע אודות הספרייה ופונקציות נוספות שלה תוכלו למצוא [באתר הרשמי.](https://data-flair.training/blogs/basic-functionality-in-pandas/)   
  
התקנה:

pip install pandas

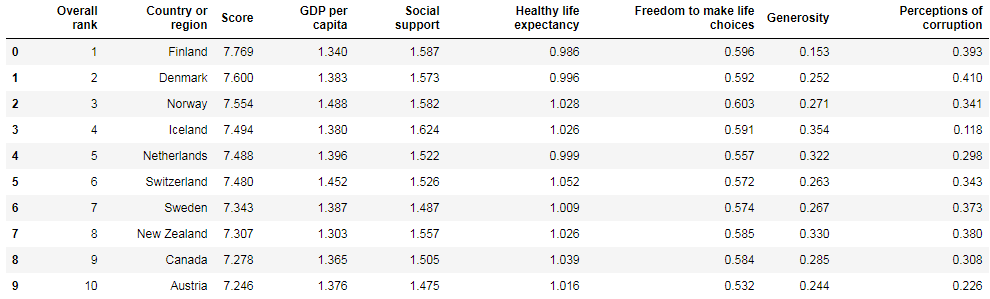
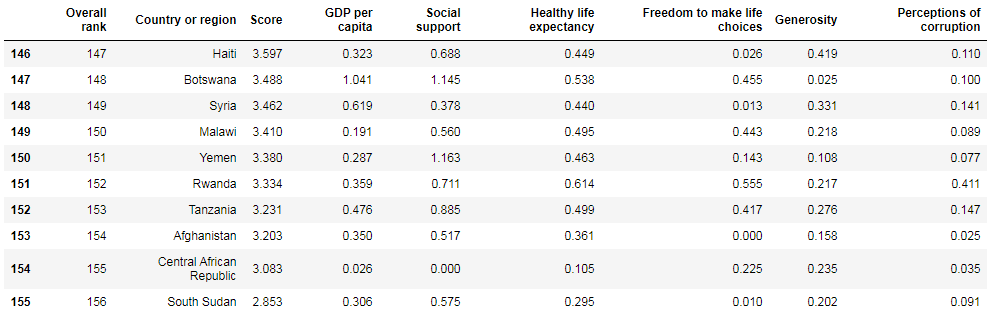
**טעינת קובץ-**   
פנדס חזקה במיוחד עבור נתונים שמרוכזים בטבלאות (נתונים רלציוניים), לרוב הנתונים שנשתמש בהם הם קבצי csv או tsv (csv- קבצים שמופרדים בפסיקים, tsv- מופרדים בטאבס).  
בחלק הקרוב נשתמש בנתונים של דו"ח האושר העולמי של שנת 2019 ,שלקוח מהאתר [kaggle](https://www.kaggle.com/unsdsn/world-happiness?select=2018.csv), אל דאגה גם ישראל מופיע שם, אם כי יש לה עוד במה להשתפר.  
בשביל לטעון קובץ csv לפנדס נשתמש בפונקציה read\_csv(str\_name)שמקבלת את שם הקובץ כפרמטר:

import pandas as pd

df = pd.read\_csv("2019.csv")

יש גם אפשרות להשתמש באותה פונקציה עם כתובת url כפרמטר במקום שם הקובץ.  
האובייקט שהתקבל מהפונקציה הוא מטיפוס data frame והוא מקביל למערך של נתונים.

לאחר שהטענו את הקובץ, פנדס מאפשרת לנו לראות חלקים מתוך הרשימה. עם הפונקציה head() נוכל לראות את ראש הרשימה, כברירת מחדל פנדס מציגה רק את חמשת הראשונים, אבל אפשר להכניס כפרמטר כמה שורות נרצה לראות.  
ובאותו האופן נוכל לראות את תחתית הרשימה עם הפונקציה tail().

df.head(10)  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
  
  
df.tail(10)  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  


בשביל לקבל קצת יותר מידע על הנתונים יש את הפונקציה info() :

df.info()  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
<bound method DataFrame.info of Overall rank Country or region Score GDP per capita \

0 1 Finland 7.769 1.340

1 2 Denmark 7.600 1.383

2 3 Norway 7.554 1.488

3 4 Iceland 7.494 1.380

4 5 Netherlands 7.488 1.396

.. ... ... ... ...

151 152 Rwanda 3.334 0.359

152 153 Tanzania 3.231 0.476

153 154 Afghanistan 3.203 0.350

154 155 Central African Republic 3.083 0.026

155 156 South Sudan 2.853 0.306

Social support Healthy life expectancy Freedom to make life choices \

0 1.587 0.986 0.596

1 1.573 0.996 0.592

2 1.582 1.028 0.603

3 1.624 1.026 0.591

4 1.522 0.999 0.557

.. ... ... ...

151 0.711 0.614 0.555

152 0.885 0.499 0.417

153 0.517 0.361 0.000

154 0.000 0.105 0.225

155 0.575 0.295 0.010

Generosity Perceptions of corruption

0 0.153 0.393

1 0.252 0.410

2 0.271 0.341

3 0.354 0.118

4 0.322 0.298

.. ... ...

151 0.217 0.411

152 0.276 0.147

153 0.158 0.025

154 0.235 0.035

155 0.202 0.091

[156 rows x 9 columns]>

כפי שאתם רואים הפונקציה מראה לנו מהם חמשת הנתונים הכי גבוהים והכי נמוכים בכל מדד.  
במקרה שלנו יש את שמות המדינות, הדירוג שלהן, הניקוד שלהן בסה"כ, תוצר לנפש, תמיכה חברתית ,תוחלת חיים בריאה  
חופש הבחירה, נדיבות ומדד תפיסת השחיתות.  
חוץ מזה יש לנו מידע על הקובץ עצמו- כמות השורות והעמודות וכו'.  
  
נניח ונרצה לראות רק עמודות ספציפיות מתוך הכלל נוכל להשתמש באינדוקס לפי שם העמודה- נשתמש בסוגריים מורבעים ובהם נכתוב רשימה של שמות העמודות שאותן נרצה לראות, למשל נרצה לראות את שם המדינה וכמה היא "נדיבה". כברירת מחדל אנחנו רואים את חמשת המקומות הראשונים והאחרונים, אבל גם כאן אפשר להשתמש בפונקציה head() ובפונקציה tail() כדי לראות את המקומות האחרונים והראשונים

df[['Country or region','Generosity']] # shows only these two columns   
df[['Country or region','Generosity']] # shows top 5 countries

אם נרצה לראות את אחת מהשורות לפי אינדקס נוכל להשתמש בפונקציה iloc() שמקבלת את האינדקס של השורה ומחזירה את השורה עצמה:

df.iloc[2]  
\_\_\_\_\_\_\_\_\_\_\_\_\_  
Overall rank 3

Country or region Norway

Score 7.554

GDP per capita 1.488

Social support 1.582

Healthy life expectancy 1.028

Freedom to make life choices 0.603

Generosity 0.271

Perceptions of corruption 0.341

Name: 2, dtype: object

עכשיו שאנחנו יודעים גם מה שמות העמודות של הקובץ נוכל לטעון אותו מחדש ולקבוע עמודה אחרת כעמודה הראשית במקום הדיפולטיבית, למשל נקבע שהעמודה של שמות המדינות היא הראשית.  
איך מעשה את זה? עם הפונקציה read\_csv() רק שנוסיף לה פרמטר index\_col עם שם העמודה כפרמטר.  
במה זה עוזר לנו? שעכשיו אנחנו יכולים לגשת ישרות למדינה שמעניינת אותנו באינדוקס ישיר עם הפונקציה loc שמשתמשת בעמודה הראשית בתור מסנן, ככה נוכל למצוא את הנתון שבאמת מעניין אותנו(ישראל):

df\_country = pd.read\_csv("2019.csv" , index\_col = 'Country or region' )  
df\_country.loc['Israel']  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Overall rank 13.000

Score 7.139

GDP per capita 1.276

Social support 1.455

Healthy life expectancy 1.029

Freedom to make life choices 0.371

Generosity 0.261

Perceptions of corruption 0.082

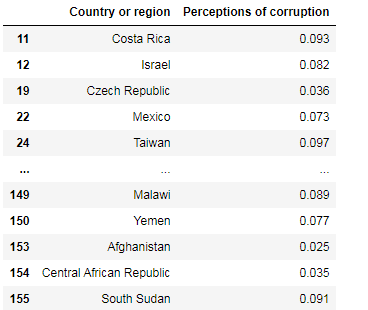
Name: Israel, dtype: float64

זה בדיוק המקום להמליץ לקרוא עוד על הדוקומנטציה של פנדס. לפנדס יש כל כך הרבה פונקציות ופרמטרים וקשה מאוד לעקוב אחריהם וזכור אותם, מומלץ בחום לעניין אולי יש פרמטרים שיותר רלוונטיים עבורכם בפרויקט ספציפי זה או אחר.

שאילתות בסיסיות-  
האופרטור '[]' של פנדס מצפה לקבל פונקציה או רשימה ממיינת, למשל כשרצינו למצוא את שמות הערים וכמה הן נדיבות הכנסו לאופרטור רשימה עם שמות העמודות שאותן רצינו.  
נניח אנחנו למצוא את כל המדינות [שמדד השחיתות](https://he.wikipedia.org/wiki/%D7%9E%D7%93%D7%93_%D7%AA%D7%A4%D7%99%D7%A1%D7%AA_%D7%94%D7%A9%D7%97%D7%99%D7%AA%D7%95%D7%AA) שלהן הוא מתחת ל-0.1 (ככל שהמדד יותר נמוך זה אומר שהמדינה יותר מושחתת), נוכל לעשות את זה ע"י סינון כפול: תחילה נרצה לקבל רק את רשימת השחיתות של המדינות, נוכל לעשות את זה באינדוקס ישיר לפי העמודה 'שחיתות' ,אח"כ נוסיף פרמטר השוואתי:

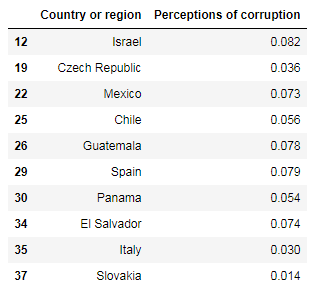
df[df['Perceptions of corruption'] < 0.1]

במקרה זה נקבל data frame חדש עם כל העמודות, לכן נוכל לסנן אותו עם סוגרים מרובעים שוב , ולהזין שם את העמודות הספציפיות שאנחנו צריכים מתוכם:

df[df['Perceptions of corruption'] < 0.1][['Country or region','Perceptions of corruption']]  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  


לא מחמיא כל כך לישראל האמת.  
טוב אז בואו נמצא את עשרת המדינות שמושחתות לפחות כמו ישראל כדי שנרגיש יותר טוב עם עצמנו:

corruptions\_countries = df[df['Perceptions of corruption']<= df\_country.loc['Israel']['Perceptions of corruption']]

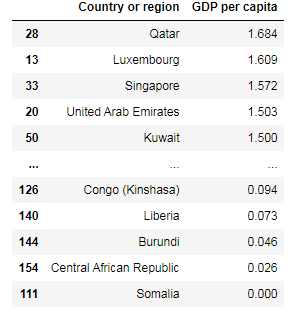
corruptions\_countries[['Country or region','Perceptions of corruption']].head(10)  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  


נניח אנחנו רוצים לשמור את ה-data frame שהקבל לקובץ csv חדש, נוכל להשתמש בפונקציה to\_csv(str\_name)   
כדי לעשות זאת, למשל נשמור את רשימת המדינות שמושחתות יותר מישראל בקובץ חדש:

corruptions\_countries.to\_csv('More corrupt than Israel.csv')

נניח אנחנו רוצים למיין את הטבלה לפי משתנה אחר ולא לפי המיון הסטנדרטי של הדירוג, למשל נרצה למיין רק לפי התוצר לנפש נוכל להשתמש בפונקציה sort\_value() ועם הפרמטר by נוכל לקבוע לפי אילו פרמטרים, ועם הפרמטר ascendind האם בסדר עולה או יורד:

df.sort\_values(by=['GDP per capita'],ascending=False) [['Country or region', 'GDP per capita']]  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



שינוי ערכים-

אם נרצה גם לשנות ערכים למשל להפוך את כל השמות לאותיות קטנות, לשנות את הפורמט של המספרים וכו' נוכל להשתמש בפונקציה apply() שמקבלת כפרמטר פונקציה ופרמטר שני axis כלומר האם לבצע את הפונקציה כעמודה או כשורה(1=שורה, הברירת מחדל כעמודה) :

def to\_lower\_case(row):

return row['Country or region'].lower()

df\_lower = df.apply(to\_lower\_case,axis=1)

df\_lower  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
0 finland

1 denmark

2 norway

3 iceland

4 netherlands

...

151 rwanda

152 tanzania

153 afghanistan

154 central african republic

155 south sudan

Length: 156, dtype: object

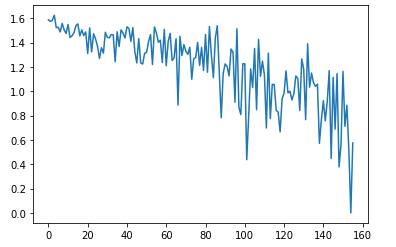
ואם נרצה להחליף את העמודה החדשה לעמודה בטבלה נשתמש באופרטור השמה:

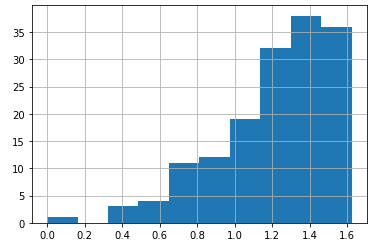
df['Country or region'] = df\_lower

ובדומה למילון, אם נרצה להוסיף עמודה חדשה לטבלה נוכל להשתמש בסוגריים מרובעים שבתוכם שם העמודה החדשה, ולהשתמש באופרטור השמה כדי לתת לה ערך.

matplotlib ו- pandas –

פנדס מספקת לנו אופציה להציג את הנתונים לפי הגדרה של עמודה וצורת תצוגה, למשל אם נרצה לראות את הנתונים של התמיכה החברתית בצורה של קו או בהיסטוגרמה נוכל להשתמש בפונקציה plot() או hist() על העמודה שנבחרה:

df['Social support'].plot()  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  


df['Social support'].hist()  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  


אפשר גם להציג כמה נתונים בבאת אחת:

df[['Generosity','Healthy life expectancy']].hist();  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
