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.

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, 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)
```

	Overall rank	Country or region	Score	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption
0	1	Finland	7.769	1.340	1.587	0.986	0.596	0.153	0.393
1	2	Denmark	7.600	1.383	1.573	0.996	0.592	0.252	0.410
2	3	Norway	7.554	1.488	1.582	1.028	0.603	0.271	0.341
3	4	Iceland	7.494	1.380	1.624	1.026	0.591	0.354	0.118
4	5	Netherlands	7.488	1.396	1.522	0.999	0.557	0.322	0.298
5	6	Switzerland	7.480	1.452	1.526	1.052	0.572	0.263	0.343
6	7	Sweden	7.343	1.387	1.487	1.009	0.574	0.267	0.373
7	8	New Zealand	7.307	1.303	1.557	1.026	0.585	0.330	0.380
8	9	Canada	7.278	1.365	1.505	1.039	0.584	0.285	0.308
9	10	Austria	7.246	1.376	1.475	1.016	0.532	0.244	0.226

```
df.tail(10)
```

	Overall rank	Country or region	Score	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption
146	147	Haiti	3.597	0.323	0.688	0.449	0.026	0.419	0.110
147	148	Botswana	3.488	1.041	1.145	0.538	0.455	0.025	0.100
148	149	Syria	3.462	0.619	0.378	0.440	0.013	0.331	0.141
149	150	Malawi	3.410	0.191	0.560	0.495	0.443	0.218	0.089
150	151	Yemen	3.380	0.287	1.163	0.463	0.143	0.108	0.077
151	152	Rwanda	3.334	0.359	0.711	0.614	0.555	0.217	0.411
152	153	Tanzania	3.231	0.476	0.885	0.499	0.417	0.276	0.147
153	154	Afghanistan	3.203	0.350	0.517	0.361	0.000	0.158	0.025
154	155	Central African Republic	3.083	0.026	0.000	0.105	0.225	0.235	0.035
155	156	South Sudan	2.853	0.306	0.575	0.295	0.010	0.202	0.091

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 עזר 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 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']]
```

	Country or region	Perceptions of corruption
11	Costa Rica	0.093
12	Israel	0.082
19	Czech Republic	0.036
22	Mexico	0.073
24	Taiwan	0.097
...	...	...
149	Malawi	0.089
150	Yemen	0.077
153	Afghanistan	0.025
154	Central African Republic	0.035
155	South Sudan	0.091

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)
```

	Country or region	Perceptions of corruption
12	Israel	0.082
19	Czech Republic	0.036
22	Mexico	0.073
25	Chile	0.056
26	Guatemala	0.078
29	Spain	0.079
30	Panama	0.054
34	El Salvador	0.074
35	Italy	0.030
37	Slovakia	0.014

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']]
```

	Country or region	GDP per capita
28	Qatar	1.684
13	Luxembourg	1.609
33	Singapore	1.572
20	United Arab Emirates	1.503
50	Kuwait	1.500
...	...	...
126	Congo (Kinshasa)	0.094
140	Liberia	0.073
144	Burundi	0.046
154	Central African Republic	0.026
111	Somalia	0.000

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:

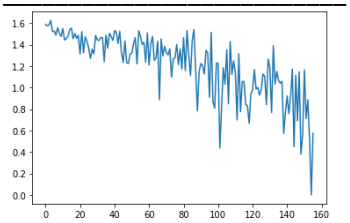
```
df['Country or region'] = df_lower
```

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

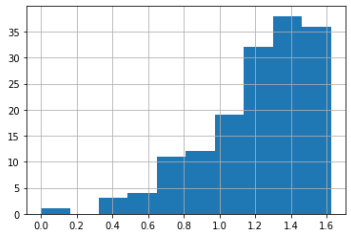
matplotlib ו- pandas –

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();
```