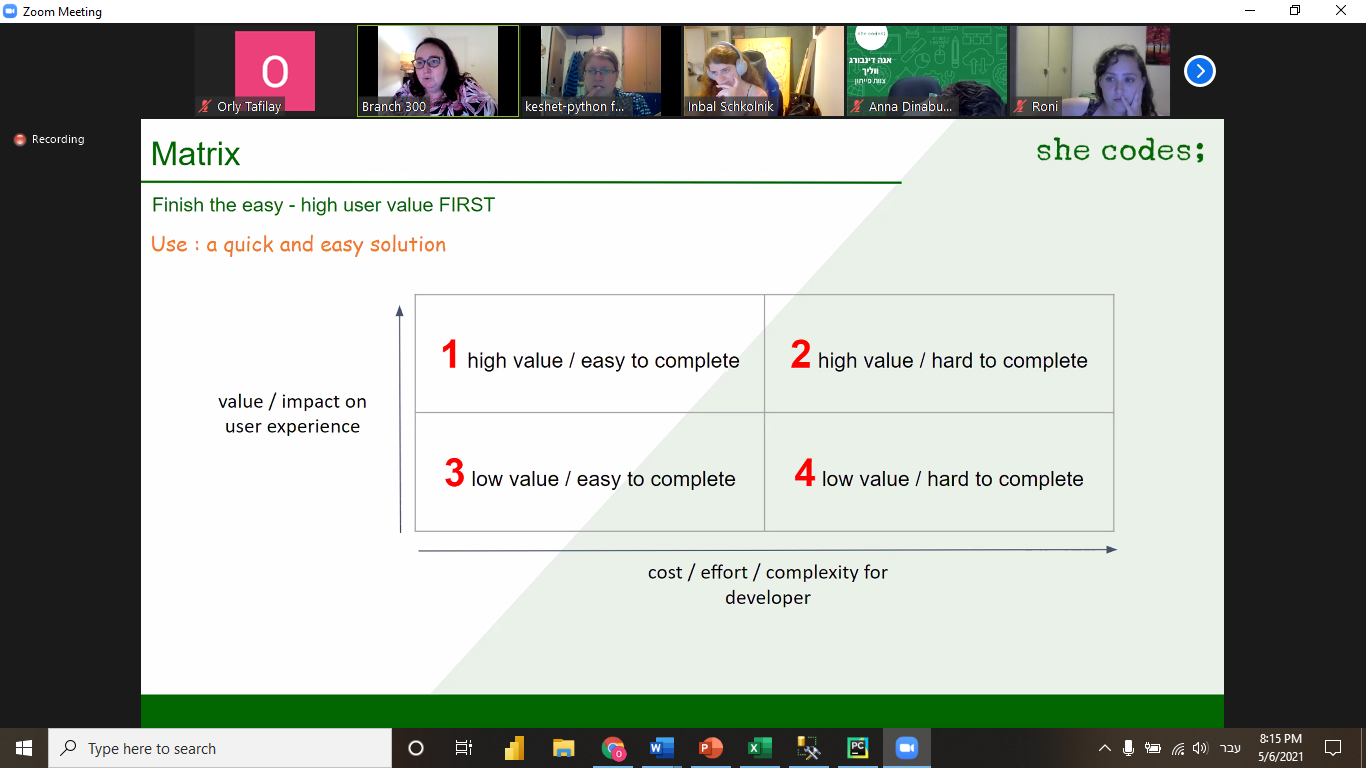
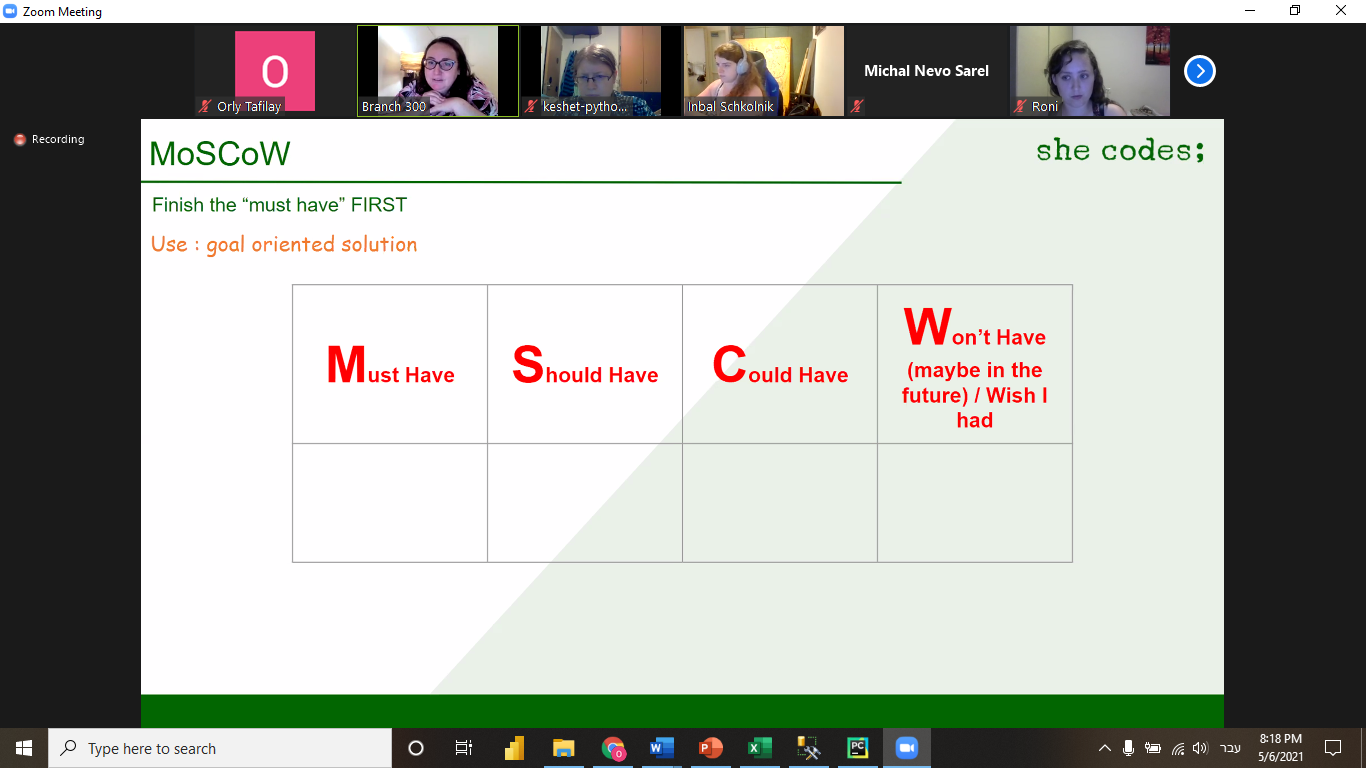
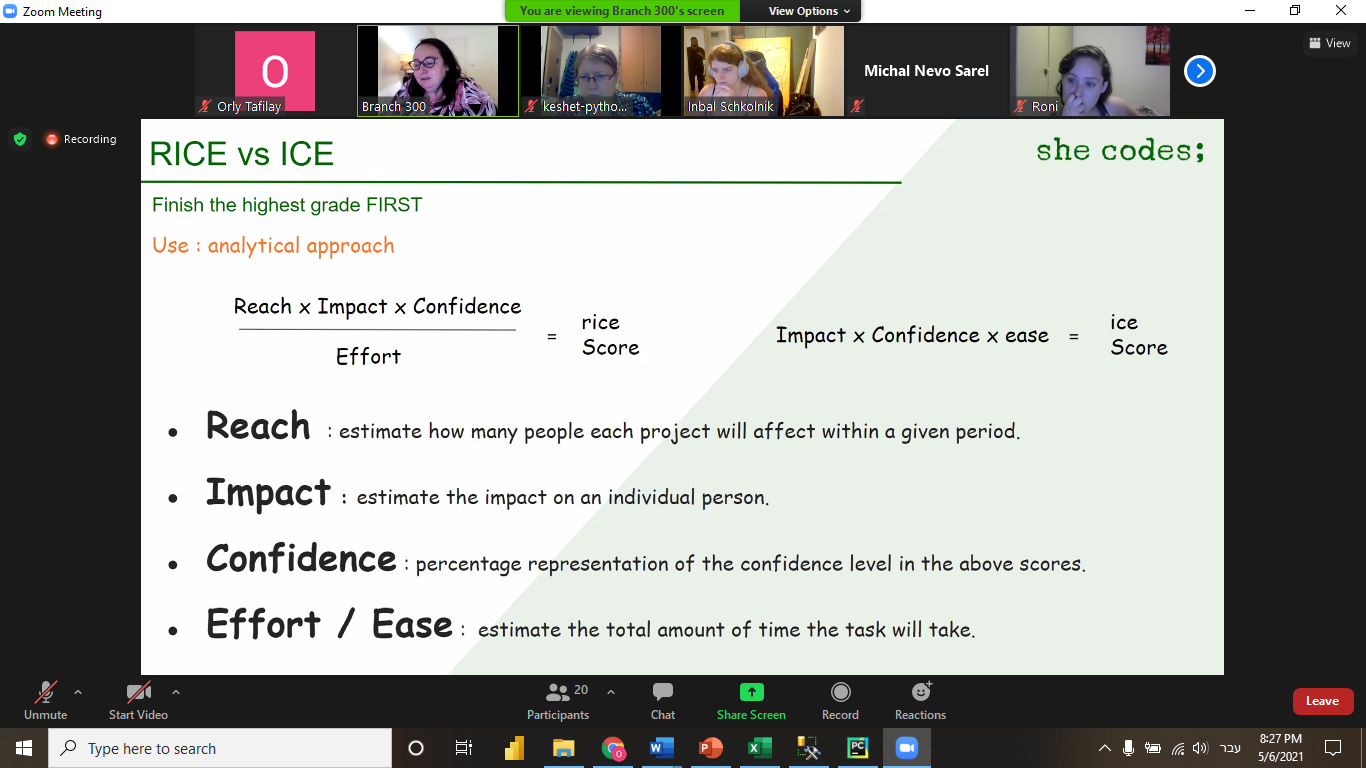
<https://www.ncdc.noaa.gov/cag/statewide/time-series/40/tavg/12/12/2020-2021?base_prd=true&begbaseyear=1901&endbaseyear=2000>  
  
https://web.archive.org/web/20150905135247/http://lebanese-economy-forum.com/wdi-gdf-advanced-data-display/show/EN-CLC-AVRT-C/

import pandas as pd  
  
# reading the data from the excel file about area size - square km  
google\_data = pd.read\_excel(r"C:\Users\User\Desktop\Final Project\Google\_data.xlsx")  
  
google\_data.columnsIndex(['country', 'area'], dtype='object')  
  
# reading the data from the excel file about Kaggle dataset  
world\_happiness = pd.read\_excel(r"C:\Users\User\Desktop\Final Project\World\_Happiness\_2021\_DB.xlsx")  
  
# merging the data from Kaggle with the data from google, about area size  
all\_data = world\_happiness.merge(google\_data, how='left', left\_on='Country name', right\_on='country')  
all\_data.drop\_duplicates(inplace=True)  
all\_data.drop(['country'], axis=1, inplace=True)  
  
Avg\_temprature = pd.read\_excel(r"C:\Users\User\Desktop\Final Project\Average\_Temp\_Celsius.xlsx")  
Avg\_temp2=pd.read\_excel(r"C:\Users\User\Desktop\Final Project\average\_temprature\_NOAA.xlsx")  
  
# using formulas to convert Fahrenheit degrees to Celsius degrees  
Avg\_temp2['AvgTemp']= Avg\_temp2['averageTemperature'].apply(lambda x:((x-32)\*(5/9)))  
Avg\_temp2.drop(['averageTemperature'],axis=1,inplace=True)  
Avg\_temprature.rename({'Country Name':'country','Average yearly temperature (Celsius)':'AvgTemp'},inplace=True, axis=1)  
all\_temprature = Avg\_temprature.append(Avg\_temp2)  
  
# reading information from excel and csv files  
avg\_precipitation = pd.read\_excel(r"C:\Users\User\Desktop\Final Project\avg\_precipitation\_depth.xlsx")  
unemploymentRate = pd.read\_excel(r"C:\Users\User\Desktop\Final Project\unemployed\_rate.xlsx")  
dis\_from\_equator = pd.read\_excel(r"C:\Users\User\Desktop\Final Project\distance\_from\_equator.xlsx")  
population = pd.read\_excel(r"C:\Users\User\Desktop\Final Project\population\_number.xlsx")  
life\_level= pd.read\_csv(r"C:\Users\User\Desktop\Final Project\life\_level.csv")  
continent= pd.read\_csv(r"C:\Users\User\Desktop\Final Project\continent.csv")  
independence\_year= pd.read\_csv(r"C:\Users\User\Desktop\Final Project\independence year.csv")  
  
# dropping cells that have no information, type: NA  
dis\_from\_equator.dropna(subset=['capital'],inplace=True)  
avg\_precipitation.dropna(subset=['avg precipitation'],inplace=True)  
continent.dropna(subset=['country'],inplace=True)  
  
# dropping unnecessary columns from dataset  
dis\_from\_equator.drop(['city'], axis=1, inplace=True)  
dis\_from\_equator.drop(['capital'], axis=1, inplace=True)  
  
# changing the location of the columns in the dataset  
dis\_from\_equator = dis\_from\_equator[['country', 'lat']]  
  
# calculating the distance from the equator with latitude coordinate, and converting it from miles to km  
dis\_from\_equator['disFromEquator'] = dis\_from\_equator['lat'].apply(lambda x:x\*69\*1.609344)  
dis\_from\_equator.drop\_duplicates(subset="country",inplace=True)  
  
# after we calculated the distance from equator, we won't need the latitude column  
dis\_from\_equator.drop(['lat'], axis=1, inplace=True)  
  
# merging the data into one table and cleaning extra country columns  
all\_data = all\_data.merge(population, how='left', left\_on='Country name', right\_on='Country Name')  
all\_data.drop(['Country Name'], axis=1, inplace=True)  
all\_data = all\_data.merge(avg\_precipitation, how='left', left\_on='Country name', right\_on='country')  
all\_data.drop(['country'], axis=1, inplace=True)  
all\_data = all\_data.merge(dis\_from\_equator, how='left', left\_on='Country name', right\_on='country')  
all\_data.drop(['country'], axis=1, inplace=True)  
all\_data = all\_data.merge(unemploymentRate, how='left', left\_on='Country name', right\_on='country')  
all\_data.drop(['country'], axis=1, inplace=True)  
all\_data = all\_data.merge(all\_temprature, how='left', left\_on='Country name', right\_on='country')  
all\_data.drop(['country'], axis=1, inplace=True)  
all\_data = all\_data.merge(continent, how='left', left\_on='Country name', right\_on='country')  
all\_data.drop(['country'], axis=1, inplace=True)  
all\_data = all\_data.merge(life\_level, how='left', left\_on='Country name', right\_on='country')  
all\_data.drop(['country'], axis=1, inplace=True)  
all\_data = all\_data.merge(independence\_year, how='left', left\_on='Country name', right\_on='country')  
all\_data.drop(['country'], axis=1, inplace=True)  
  
life\_level['country'] = life\_level['country'].apply(lambda x: x[4:])  
  
# defining function for life level categorization  
def func(standard\_of\_living):  
 if (standard\_of\_living>=0.8):  
 return 'good'  
 if (standard\_of\_living<0.8 and standard\_of\_living>=0.6):  
 return 'medium'  
 else:  
 return 'bad'  
  
life\_level['standard\_of\_living'] = life\_level['standard\_of\_living'].apply(func)  
  
# converting command from csv file wuth special string \xa0  
life\_level['country'] = life\_level['country'].str.replace('\xa0', '').astype(str)  
  
# changing the order of the columns in the main dataset  
all\_data = all\_data[['Country name','continent','Regional indicator','Ladder score','Standard error of ladder score','upperwhisker','lowerwhisker','Logged GDP per capita','Social support','Healthy life expectancy','Freedom to make life choices','Generosity','Perceptions of corruption','Ladder score in Dystopia','Explained by: Log GDP per capita','Explained by: Social support','Explained by: Healthy life expectancy','Explained by: Freedom to make life choices','Explained by: Generosity','Explained by: Perceptions of corruption','Dystopia + residual','area','population (thousands)','avg precipitation','disFromEquator','unemployed\_rate\_2020','AvgTemp']]  
  
# saving all\_data df to csv file  
all\_data.to\_csv(r"C:\Users\User\Desktop\Final Project\all\_data.csv")



- הגדרת קיי פי איי 1  
2- לכתוב קוד ולנקות את הנתונים באמצעות פייתון פנדה

3- לסדר את הקוד  
4-



**הקדמה:**  
הפרויקט שבחרתי לעבוד עליו קשור לדרך שבה ניתן למדוד אושר במדינות שונות בעולם. מטרת הפרויקט היא לסקור את רמת האושר הקיימת במדינות השונות ולהראות כיצד ***"מדע חדש"*** (מדע האושר) מסביר סוגים שונים של אושר, הן מבחינה אישית והן מבחינה מדינית. מקור המידע הראשוני שלי עוסק בדו"ח שצובר הכרה אצל ממשלות וארגונים ציבוריים רבים, שמטרתם לבדוק כיצד מדדי האושר משפיעים על מדיניות קבלת ההחלטות. הדו"ח מציין מספר פרמטרים שנבדקו על ידי חוקרים מובילים בתחומי הכלכלה, הפסיכולוגיה, הבריאות והמדינה, ונמצא כי הם יכולים להוות אינדיקציה לקדמה ולפריצת דרך בתחום.

Content

The happiness scores and rankings use data from the Gallup World Poll . The columns following the happiness score estimate the extent to which each of six factors – economic production, social support, life expectancy, freedom, absence of corruption, and generosity – contribute to making life evaluations higher in each country than they are in Dystopia, a hypothetical country that has values equal to the world’s lowest national averages for each of the six factors. They have no impact on the total score reported for each country, but they do explain why some countries rank higher than others.  
  
**טרמינולוגיה:**  
Database  
Dataset  
Query  
Pandas Python