Amany Mohammed Al Luhaybi

amluhaybi@uqu.edu.sa

MVP for (Predicting the ratings of Coursera course reviews)

Link (https://github.com/Amany0/Project1/blob/master/Code.ipynb**)**

Basic Data Exploration and Data Cleaning

- **Merge two datasets the reviews & courses on the column 'course_id' get one dataframe**

```
courses = pd.read_csv('Coursera_courses.csv')
reviews = pd.read_csv('Coursera_reviews.csv')
```

```
# merge two datasets the reviews & courses  on the column 'course_id' get one dataframe
df = pd.merge(reviews, courses, on='course_id')
df
```

	reviews	reviewers	date_reviews	rating	course_id	name	institution	course_url
0	Pretty dry, but I was able to pass with just t...	By Robert S	Feb 12, 2020	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	https://www.coursera.org/learn/google-cbrs-cpi...
1	would be a better experience if the video and ...	By Gabriel E R	Sep 28, 2020	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	https://www.coursera.org/learn/google-cbrs-cpi...
2	Information was perfect! The program itself wa...	By Jacob D	Apr 08, 2020	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	https://www.coursera.org/learn/google-cbrs-cpi...
3	A few grammatical mistakes on test made me do ...	By Dale B	Feb 24, 2020	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	https://www.coursera.org/learn/google-cbrs-cpi...

- **The shape, head, info, describe, of the dataset and create new columns**

```
df.head()
```

	reviews	reviewers	date_reviews	rating	course_id	name	institution	course_url
0	Pretty dry, but I was able to pass with just t...	By Robert S	Feb 12, 2020	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	https://www.coursera.org/learn/google-cbrs-cpi...
1	would be a better experience if the video and ...	By Gabriel E R	Sep 28, 2020	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	https://www.coursera.org/learn/google-cbrs-cpi...
2	Information was perfect! The program itself wa...	By Jacob D	Apr 08, 2020	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	https://www.coursera.org/learn/google-cbrs-cpi...
3	A few grammatical mistakes on test made me do ...	By Dale B	Feb 24, 2020	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	https://www.coursera.org/learn/google-cbrs-cpi...
4	Excellent course and the training provided was...	By Sean G	Jun 18, 2020	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	https://www.coursera.org/learn/google-cbrs-cpi...

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1454711 entries, 0 to 1454710
Data columns (total 8 columns):
 #   Column        Non-Null Count    Dtype
---  ------        --------------    -----
 0   reviews       1454571 non-null  object
 1   reviewers     1454711 non-null  object
```

```
df.describe()
```

	rating
count	1.454711e+06
mean	4.696649e+00
std	6.983271e-01
min	1.000000e+00
25%	5.000000e+00

```
#number of rows- reviews-
df.shape
```

```
(1454711, 8)
```

```
# create new columns of month & year to replace the date_reviews column.
df['year'] = pd.to_datetime(df['date_reviews']).dt.year
df['day'] = pd.to_datetime(df['date_reviews']).dt.day
df['month'] = pd.to_datetime(df['date_reviews']).dt.month

df = df.drop('date_reviews', 1)
```

```
df
```

	reviews	reviewers	rating	course_id	name	institution	course_url	year	day	month
0	Pretty dry, but I was able to pass with just t...	By Robert S	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	https://www.coursera.org/learn/google-cbrs-cpi...	2020	12	2
1	would be a better experience if the video and ...	By Gabriel E R	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	https://www.coursera.org/learn/google-cbrs-cpi...	2020	28	9

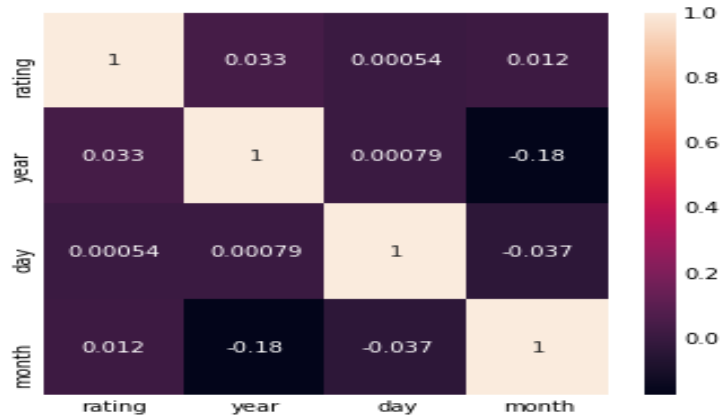
- **drop course URL column**

```
#drop course URL column
df = df.drop('course_url', 1)
df
```

	reviews	reviewers	rating	course_id	name	institution	year	day	month
0	Pretty dry, but I was able to pass with just t...	By Robert S	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	2020	12	2
1	would be a better experience if the video and ...	By Gabriel E R	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	2020	28	9

- **Finding the correlation between the feature column**

```
plt.figure(figsize=(5,5))
sns.heatmap(df.corr(), annot=True)
plt.show()
```



- **find out the total number of missing values in each column**

```
#find out the total number of missing values in each column
df.isnull().sum()
```

```
reviews        140
reviewers        0
rating           0
course_id        0
name             0
institution      0
year             0
day              0
month            0
dtype: int64
```

- **there is null values in reviews column I will check the rating if it was high then I will fill the null value with 'good high rate; otherwise it will be filled with bad low rate**

```
# there is null values in reviews column I will check the rating if it was high
#then I will fill the null value with 'good high rate; otherwise it will be filled with bad low rate

mask= df[df.reviews.isnull()]


mask['reviews']=np.where(mask['rating'] >= 3, 'good high rate', 'bad low rate')
df['reviews'].fillna(mask['reviews'], inplace=True)
df.head(1000)
```

	reviews	reviewers	rating	course_id	name	institution	year	day	month
0	Pretty dry, but I was able to pass with just t...	By Robert S	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	2020	12	2
1	would be a better experience if the video and ...	By Gabriel E R	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	2020	28	9

- **examining the datafram after dealing with null values**

```
#examining the datafram after dealing with null values
df.isnull().sum()
```

```
reviews        0
reviewers      0
rating         0
course_id      0
name           0
institution    0
year           0
day            0
month          0
dtype: int64
```

- **create a new column sentiment which will be used later as label for text classification**

```
# Set ratings >3  to 1, the rest to 0
df['sentiment'] = np.where(df['rating'] >= 3, '1', '0')
df.head()
```

	reviews	reviewers	rating	course_id	name	institution	year	day	month	sentiment
0	Pretty dry, but I was able to pass with just t...	By Robert S	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	2020	12	2	1
1	would be a better experience if the video and ...	By Gabriel E R	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	2020	28	9	1

- **check for duplicated rows and drop them**

```
df.duplicated().sum()
```

```
934764
```

```
#check for duplicated rows

df.duplicated(subset=["course_id","reviewers","reviews","name"]).value_counts()
```

```
True     934783
False    519928
dtype: int64
```

```
df.drop_duplicates(subset=["course_id","reviewers","reviews","name"], keep='first', inplace=True)
df
```

	reviews	reviewers	rating	course_id	name	institution	year	day	month	sentiment
0	Pretty dry, but I was able to pass with just t...	By Robert S	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	2020	12	2	1
1	would be a better experience if the video and ...	By Gabriel E R	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	2020	28	9	1

- **shape of dataframe after dropping the duplicates**

```
df.shape
```

```
(519928, 10)
```

- **cleaning the reviews column as it will be used later on text classification**

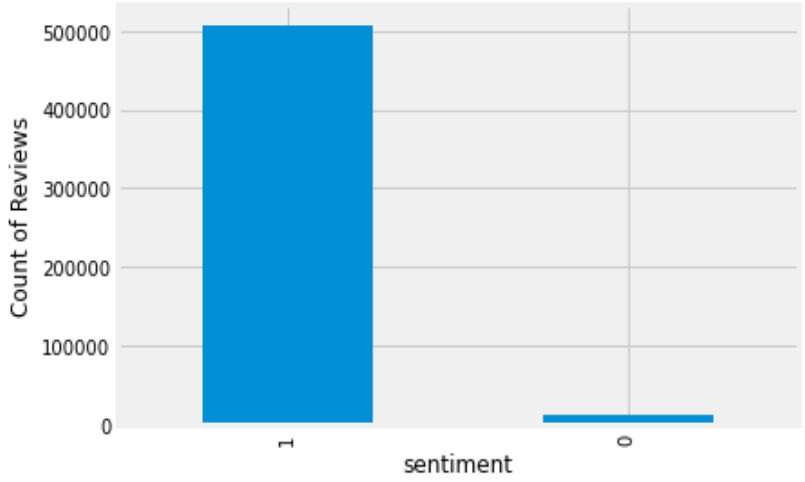
```
# Text preprocessing steps - remove numbers, captial letters and punctuation
alphanumeric = lambda x: re.sub('\w*\d\w*', ' ', x)
punc_lower = lambda x: re.sub('[%s]' % re.escape(string.punctuation), ' ', x.lower())


df['reviews'] = df.reviews.map(alphanumeric).map(punc_lower)
df.head()
```

	reviews	reviewers	rating	course_id	name	institution	year	day	month	sentiment
0	pretty dry but i was able to pass with just t...	By Robert S	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	2020	12	2	1
1	would be a better experience if the video and ...	By Gabriel E R	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	2020	28	9	1
2	information was perfect the program itself wa...	By Jacob D	4	google-cbrs-cpi-training	Become a CBRS Certified Professional Installer...	Google - Spectrum Sharing	2020	8	4	1

- **discovering imbalance in dataset will be dealt with in many ways in the next phase**

```
#discovering imbalance in dataset
ax = df.sentiment.value_counts().plot(kind='bar')
plt.xlabel("sentiment")
plt.ylabel("Count of Reviews")
# the data is Highly imbalanced will be dealt with in the final submission
```

```
Text(0, 0.5, 'Count of Reviews')
```




```
#discovering imbalance in dataset
df['sentiment'].value_counts()
```

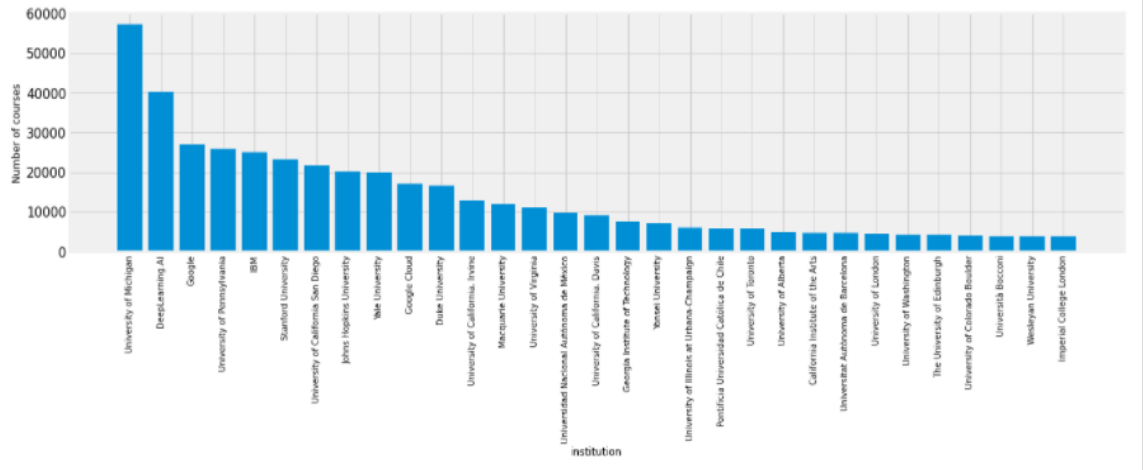
```
1    507233
0     12695
Name: sentiment, dtype: int64
```

- **Which institution had the most courses?**

```
# Which institution had the most courses?
inst_high_Courses = df.institution.value_counts()
inst_high_Courses
```

```
University of Michigan                                  57194
DeepLearning.AI                                         40329
Google                                                  26936
University of Pennsylvania                              25763
IBM                                                     24975
                                                        ...
Google - Spectrum Sharing                                  33
Peter the Great St. Petersburg Polytechnic University      33
```

```
# Using some institution because the size of data is huge
#we can see that University of Michigan has the hightes number of courses
y = df.institution.value_counts().values[0:31]
x = df.institution.value_counts().index[0:31]
plt.figure(figsize=(20,5))
plt.bar(x,y)
plt.xlabel("institution")
plt.ylabel("Number of courses")
plt.xticks(rotation='vertical',size=10)
plt.yticks(size=15)
plt.show()
```

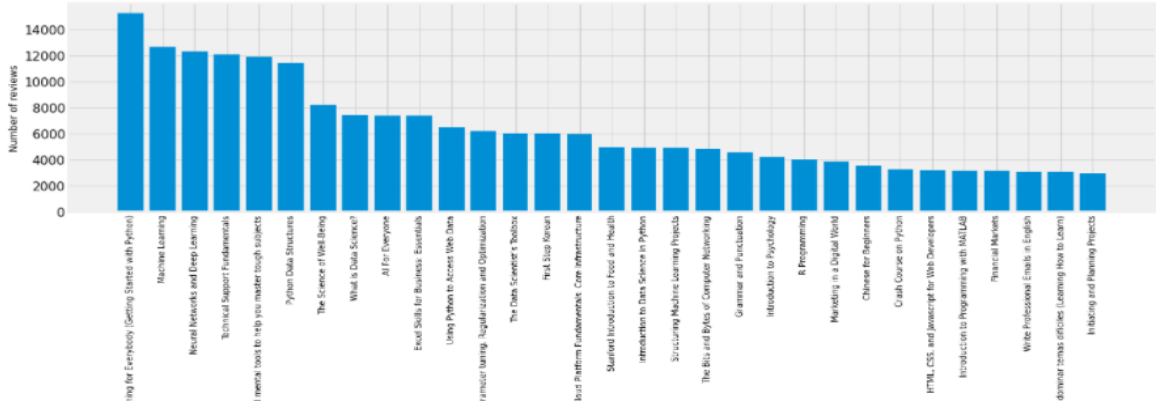


- **Which course had the highest number of reviews?**

```
#Which course had the highest number of reviews?
top_reviewed_courses = df.name.value_counts()
top_reviewed_courses.head(10)
```

```
Programming for Everybody (Getting Started with Python)                          15226
Machine Learning                                                                 12677
Neural Networks and Deep Learning                                                12292
Technical Support Fundamentals                                                   12055
Learning How to Learn: Powerful mental tools to help you master tough subjects   11871
Python Data Structures                                                           11421
The Science of Well-Being                                                         8199
What is Data Science?                                                             7397
AI For Everyone                                                                   7386
Excel Skills for Business: Essentials                                             7377
Name: name, dtype: int64
```

```
# Using some courses because the size of data is huge
#we can see that Programming for Everybody course has the hightes number of reviews
y = top_reviewed_courses.values[0:31]
x = top_reviewed_courses.index[0:31]
plt.figure(figsize=(20,5))
plt.bar(x,y)
plt.xlabel("Courses")
plt.ylabel("Number of reviews")
plt.xticks(rotation='vertical',size=10)
plt.yticks(size=15)
plt.show()
```

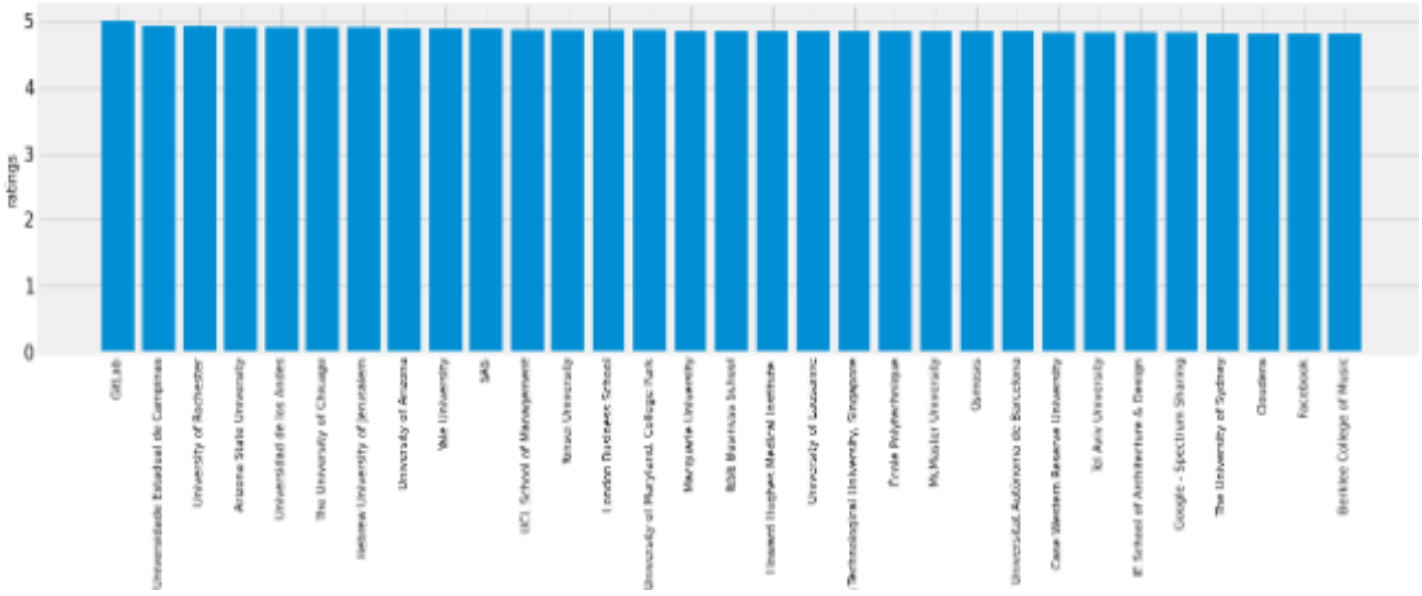


- **Which institution had the highest rates by reviewers?**

```
#Which institution had the highest rates by reviewers
top_rated_inst=df.groupby(['institution'])['rating'].mean().sort_values(ascending  = False)
top_rated_inst
```

```
institution
GitLab                                5.000000
Universidade Estadual de Campinas     4.921109
University of Rochester               4.921053
Arizona State University              4.903518
Universidad de los Andes              4.898058
                                          ...
Novosibirsk State University          4.077922
Yandex                                3.484127
New York Institute of Finance         3.463768
Saint Petersburg State University     3.387500
University of New Mexico              1.000000
Name: rating, Length: 132, dtype: float64
```

```
#Which institution had the highest rates by reviewers? GitLab
y = top_rated_inst.values[0:31]
x = top_rated_inst.index[0:31]
plt.figure(figsize=(20,5))
plt.bar(x,y)
plt.xlabel("institution")
plt.ylabel("ratings")
plt.xticks(rotation='vertical',size=10)
plt.yticks(size=15)
plt.show()
```

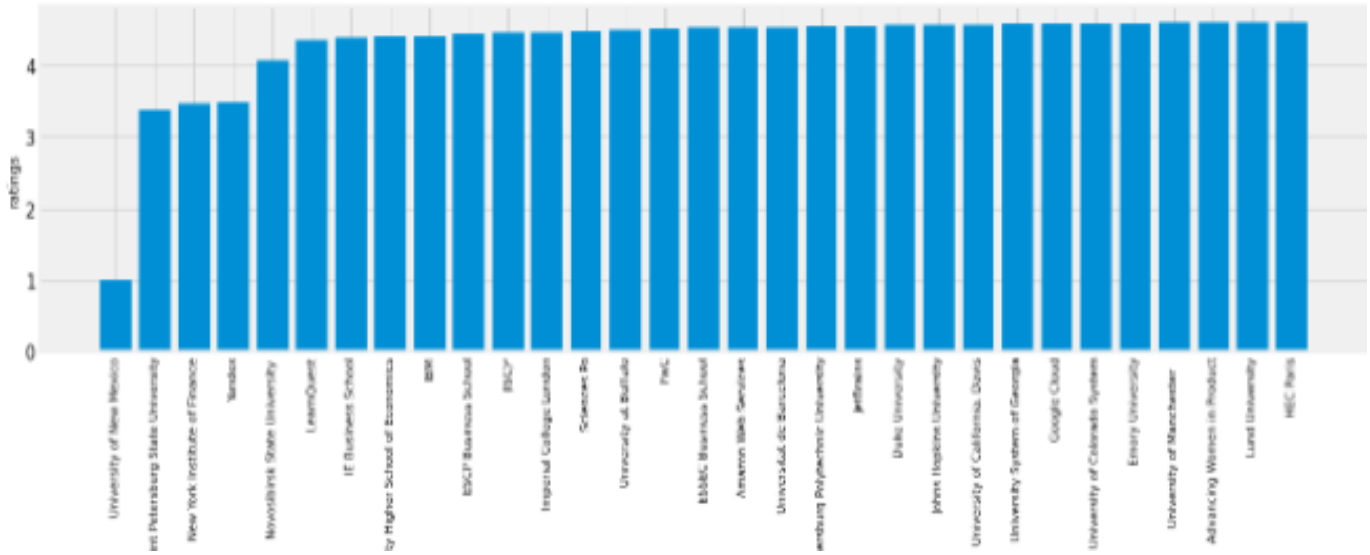


- **Which institution had the lowest rates by reviewers?**

```
#Which institution had the lowest rates by reviewers
low_rated_inst=df.groupby(['institution'])['rating'].mean().sort_values(ascending  = True )
low_rated_inst
```

```
institution
University of New Mexico             1.000000
Saint Petersburg State University    3.387500
New York Institute of Finance        3.463768
Yandex                               3.484127
Novosibirsk State University         4.077922
                                       ...
Universidad de los Andes             4.898058
Arizona State University             4.903518
University of Rochester              4.921053
Universidade Estadual de Campinas    4.921109
GitLab                               5.000000
Name: rating, Length: 132, dtype: float64
```

```
#Which institution had the lowest rates by reviewers? University of New Mexico
y = low_rated_inst.values[0:31]
x = low_rated_inst.index[0:31]
plt.figure(figsize=(20,5))
plt.bar(x,y)
plt.xlabel("institution")
plt.ylabel("ratings")
plt.xticks(rotation='vertical',size=10)
plt.yticks(size=15)
plt.show()
```

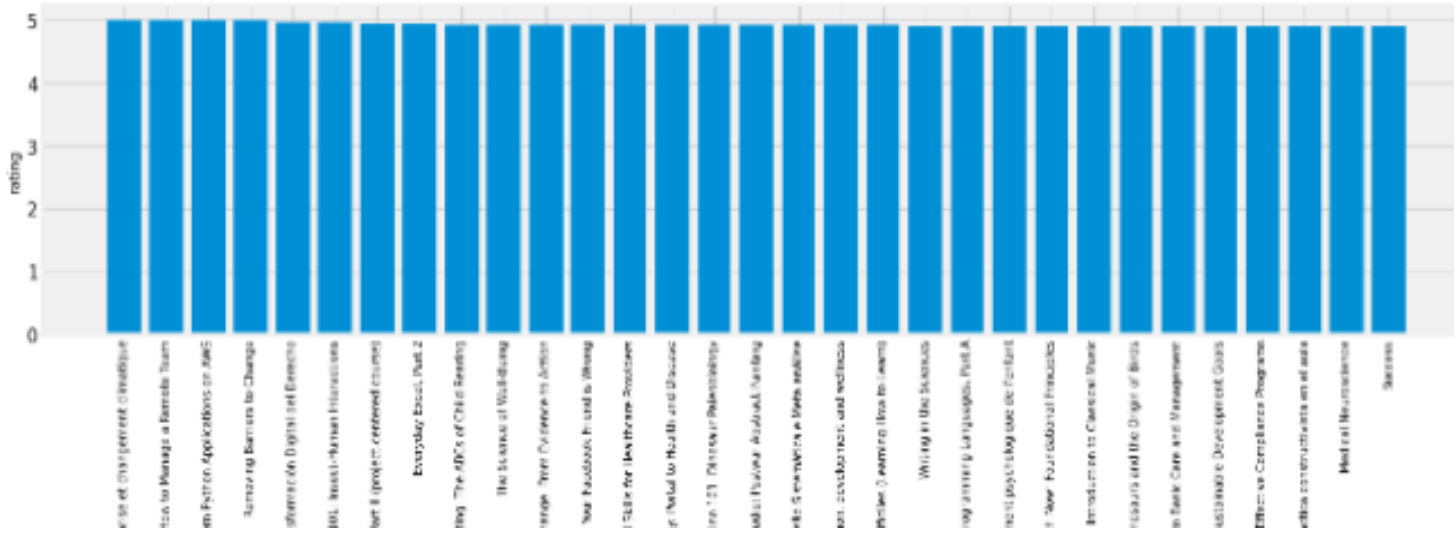


- **Which course had the highest rates by reviewers?**

```
#Which course had the highest rates by reviewers?
high_rated_courses=df.groupby(['name'])['rating'].mean().sort_values(ascending  = False)

high_rated_courses
```

```
name
Entreprise et changement climatique                                    5.000000
How to Manage a Remote Team                                            5.000000
Building Modern Python Applications on AWS                             5.000000
Removing Barriers to Change                                            5.000000
El Abogado del Futuro: Legaltech y la Transformación Digital del Derecho    4.968000
                                                                           ...
The Introduction to Quantum Computing                                  3.204545
Epigenetic Control of Gene Expression                                  1.000000
Entrepreneurship Strategy: From Ideation to Exit                       1.000000
Social and Economic Networks:  Models and Analysis                     1.000000
Curanderismo: Traditional Healing Using Plants                         1.000000
Name: rating, Length: 603, dtype: float64
```

```
#Which course had the highest rates by reviewers?
y = high_rated_courses.values[0:31]
x = high_rated_courses.index[0:31]
plt.figure(figsize=(20,5))
plt.bar(x,y)
plt.xlabel("courses")
plt.ylabel("rating")
plt.xticks(rotation='vertical',size=10)
plt.yticks(size=15)
plt.show()
```

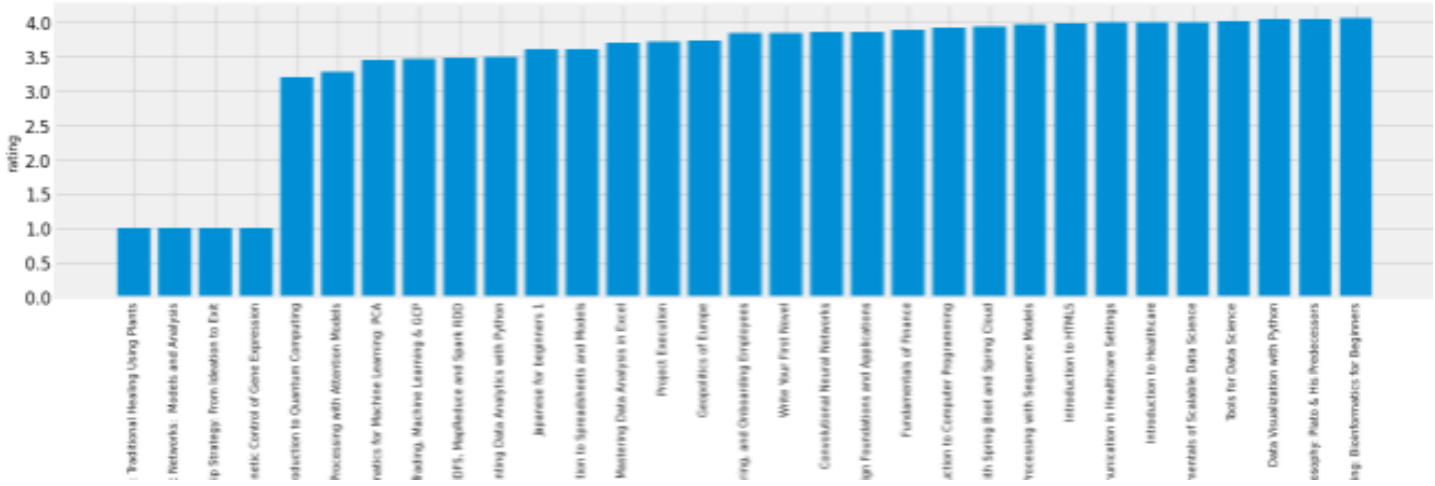


- **Which course had the lowest rates by reviewers?**

```
#Which course had the lowest rates by reviewers?
low_rated_courses=df.groupby(['name'])['rating'].mean().sort_values(ascending  = True)

low_rated_courses
```

```
name
Curanderismo: Traditional Healing Using Plants                          1.000000
Social and Economic Networks:  Models and Analysis                      1.000000
Entrepreneurship Strategy: From Ideation to Exit                        1.000000
Epigenetic Control of Gene Expression                                   1.000000
The Introduction to Quantum Computing                                   3.204545
                                                                           ...
El Abogado del Futuro: Legaltech y la Transformación Digital del Derecho    4.968000
Building Modern Python Applications on AWS                              5.000000
Entreprise et changement climatique                                     5.000000
Removing Barriers to Change                                             5.000000
How to Manage a Remote Team                                             5.000000
Name: rating, Length: 603, dtype: float64
```

```
#Which course had the lowest rates by reviewers?
y = low_rated_courses.values[0:31]
x = low_rated_courses.index[0:31]
plt.figure(figsize=(20,5))
plt.bar(x,y)
plt.xlabel("courses")
plt.ylabel("rating")
plt.xticks(rotation='vertical',size=10)
plt.yticks(size=15)
plt.show()
```



- **preparing the dataset for the text classification problem**

```
# Since I will predict the rating based on the reviews I will only need sentiment and reviews columns to start with nlp
#preparing the dataset for the text classification problem
#reviews already been processed and cleaned
data = df[['sentiment', 'reviews']]
data.head()
```

	sentiment	reviews
0	1	pretty dry but i was able to pass with just t...
1	1	would be a better experience if the video and ...
2	1	information was perfect the program itself wa...
3	1	a few grammatical mistakes on test made me do ...
4	1	excellent course and the training provided was...

```
data.shape
```

```
(519928, 2)
```