World University Rankings 2023

Welcome to the "Exploring the World's Top Universities in 2023" Jupyter notebook project. In this project, we delve into the fascinating world of higher education and academia to analyze the rankings of universities from around the globe for the year 2023.

The landscape of higher education is continually evolving, with universities competing for recognition in various academic fields. Using the latest data available, we will explore and visualize the rankings of universities based on different criteria such as academic performance, research output, international diversity, and more.

Import Library

In [1]:
```
import pandas as pd
```

In [2]:
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import seaborn as sns
```

```
C:\Users\Syed Arif\anaconda3\lib\site-packages\scipy\__init__.py:146: UserWarnin
g: A NumPy version >=1.16.5 and <1.23.0 is required for this version of SciPy (d
etected version 1.25.1
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion}"
```

Uploading Csv fle

In [3]:
```
df = pd.read_csv(r"C:\Users\Syed Arif\Desktop\World University Rankings 2023.csv"
```

Data Preprocessing

.head()

head is used show to the By default = 5 rows in the dataset

In [4]:
```
df.head()
```

Out[4]:

	University Rank	Name of University	Location	No of student	No of student per staff	International Student	Female:Male Ratio	OverAll Score	Teachin Scor
0	1	University of Oxford	United Kingdom	20,965	10.6	42%	48 : 52	96.4	92.
1	2	Harvard University	United States	21,887	9.6	25%	50 : 50	95.2	94.
2	3	University of Cambridge	United Kingdom	20,185	11.3	39%	47 : 53	94.8	90.
3	3	Stanford University	United States	16,164	7.1	24%	46 : 54	94.8	94.
4	5	Massachusetts Institute of Technology	United States	11,415	8.2	33%	40 : 60	94.2	90.

.tail()

tail is used to show rows by Descending order

In [5]:
```
df.tail()
```

Out[5]:

	University Rank	Name of University	Location	No of student	No of student per staff	International Student	Female:Male Ratio	OverAll Score	Tea
2336	-	University of the West of Scotland	NaN	NaN	NaN	NaN	NaN	34.0–39.2	
2337	-	University of Windsor	NaN	NaN	NaN	NaN	NaN	34.0–39.2	
2338	-	University of Wolverhampton	NaN	NaN	NaN	NaN	NaN	34.0–39.2	
2339	-	University of Wuppertal	NaN	NaN	NaN	NaN	NaN	34.0–39.2	
2340	-	Xi'an Jiaotong-Liverpool University	NaN	NaN	NaN	NaN	NaN	34.0–39.2	

.shape

It show the total no of rows & Column in the dataset

In [6]:
```
df.shape
```

Out[6]:
```
(2341, 13)
```

.Columns

It show the no of each Column

In [7]:
```
df.columns
```

Out[7]:
```
Index(['University Rank', 'Name of University', 'Location', 'No of student',
       'No of student per staff', 'International Student', 'Female:Male Ratio',
       'OverAll Score', 'Teaching Score', 'Research Score', 'Citations Score',
       'Industry Income Score', 'International Outlook Score'],
      dtype='object')
```

.dtypes

This Attribute show the data type of each column

In [8]:
```
df.dtypes
```

Out[8]:
```
University Rank               object
Name of University            object
Location                      object
No of student                 object
No of student per staff      float64
International Student         object
Female:Male Ratio             object
OverAll Score                 object
Teaching Score               float64
Research Score               float64
Citations Score              float64
Industry Income Score        float64
International Outlook Score  float64
dtype: object
```

.unique()

In a column, It show the unique value of specific column.

In [9]:
```
df["Location"].unique()
```

Out[9]:
```
array(['United Kingdom', 'United States', 'Switzerland', nan, 'Canada',
       'Australia', 'Singapore', 'Japan', 'France', 'Sweden', 'China',
       'South Korea', 'Netherlands', 'Germany', 'Belgium', 'Finland',
       'Denmark', 'Austria', 'Norway', 'New Zealand', 'Spain', 'Italy',
       'Saudi Arabia', 'Luxembourg', 'Qatar', 'Brazil', 'Israel',
       'Ireland', 'Taiwan', 'India', 'United Arab Emirates',
       'Brunei Darussalam', 'Iceland', 'Lebanon', 'Philippines',
       'Portugal', 'Iran', 'Malaysia', 'Poland', 'Egypt', 'Turkey',
       'Greece', 'Vietnam', 'Algeria', 'Nigeria', 'Tanzania', 'Chile',
       'Pakistan', 'Ukraine', 'Romania', 'Czech Republic', 'South Africa',
       'Northern Cyprus', 'Hong Kong', 'Ethiopia', 'Jordan', 'Serbia',
       'Sri Lanka', 'Jamaica', 'Zambia', 'Iraq', 'Costa Rica', 'Cyprus',
       'Bangladesh', 'Mozambique', 'Colombia', 'Kenya', 'Namibia', 'Peru',
       'Latvia', 'Oman', 'Thailand', 'Lithuania', 'Slovenia', 'Uganda',
       'Malta', 'Nepal', 'Kazakhstan', 'Mexico', 'Botswana', 'Slovakia',
       'Ghana', 'Morocco', 'Georgia', 'Tunisia', 'Mauritius', 'Hungary',
       'Puerto Rico', 'Ecuador', 'Fiji', 'Croatia', 'Estonia', 'Zimbabwe',
       'Indonesia', 'Argentina', 'Bulgaria', 'Venezuela', 'Azerbaijan',
       'Cuba', 'Montenegro', 'Uzbekistan', 'Palestine', 'Kuwait',
       'Somalia', 'Libya', 'Moldova', 'Kyrgyzstan', 'Malawi', 'Paraguay',
       'Mongolia', 'Armenia', 'Sudan', 'Turkmenistan', 'Uruguay',
       'Albania', 'Cambodia', 'Kosovo'], dtype=object)
```

.nuique()

It will show the total no of unque value from whole data frame

In [10]:
```
df.nunique()
```

Out[10]:
```
University Rank               162
Name of University           2233
Location                      116
No of student                2136
No of student per staff       420
International Student          79
Female:Male Ratio              87
OverAll Score                 160
Teaching Score                437
Research Score                477
Citations Score               820
Industry Income Score         391
International Outlook Score   694
dtype: int64
```

.describe()

It show the Count, mean , median etc

In [11]:
```
df.describe()
```

Out[11]:

	No of student per staff	Teaching Score	Research Score	Citations Score	Industry Income Score	International Outlook Score
count	2208.000000	1799.000000	1799.000000	1799.000000	1799.000000	1799.000000
mean	19.000408	27.018010	23.016898	48.495887	47.104558	46.880378
std	12.132224	13.282243	16.763819	27.967185	15.093682	22.582401
min	0.400000	11.600000	7.400000	0.800000	36.900000	14.100000
25%	12.600000	18.000000	11.300000	23.100000	37.800000	27.900000
50%	16.600000	22.700000	17.000000	47.200000	40.500000	42.100000
75%	22.200000	31.850000	28.900000	72.350000	48.300000	62.100000
max	232.200000	94.800000	99.700000	100.000000	100.000000	99.700000

.value_counts

It Shows all the unique values with their count

In [12]:
```
df["Location"].value_counts()
```

Out[12]:
```
United States     173
Japan             150
United Kingdom    149
India              91
China              82
                 ...
Puerto Rico         1
Mozambique          1
Mauritius           1
Namibia             1
Kosovo              1
Name: Location, Length: 116, dtype: int64
```

.isnull()

It shows the how many null values

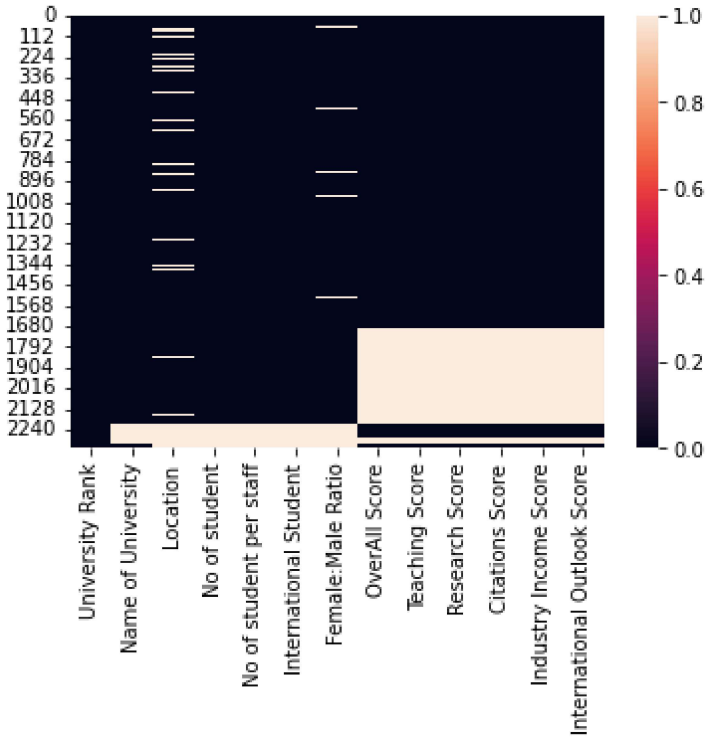
In [13]:
```
df.isnull()
```

Out[13]:

	University Rank	Name of University	Location	No of student	No of student per staff	International Student	Female:Male Ratio	OverAll Score	Teachin Scor
0	False	False	False	False	False	False	False	False	Fals
1	False	False	False	False	False	False	False	False	Fals
2	False	False	False	False	False	False	False	False	Fals
3	False	False	False	False	False	False	False	False	Fals
4	False	False	False	False	False	False	False	False	Fals
...	...	...	...	...	...	...	...	...	.
2336	False	False	True	True	True	True	True	False	Fals
2337	False	False	True	True	True	True	True	False	Fals
2338	False	False	True	True	True	True	True	False	Fals
2339	False	False	True	True	True	True	True	False	Fals
2340	False	False	True	True	True	True	True	False	Fals

2341 rows × 13 columns

In [14]:
```
sns.heatmap(df.isnull())
plt.show()
```

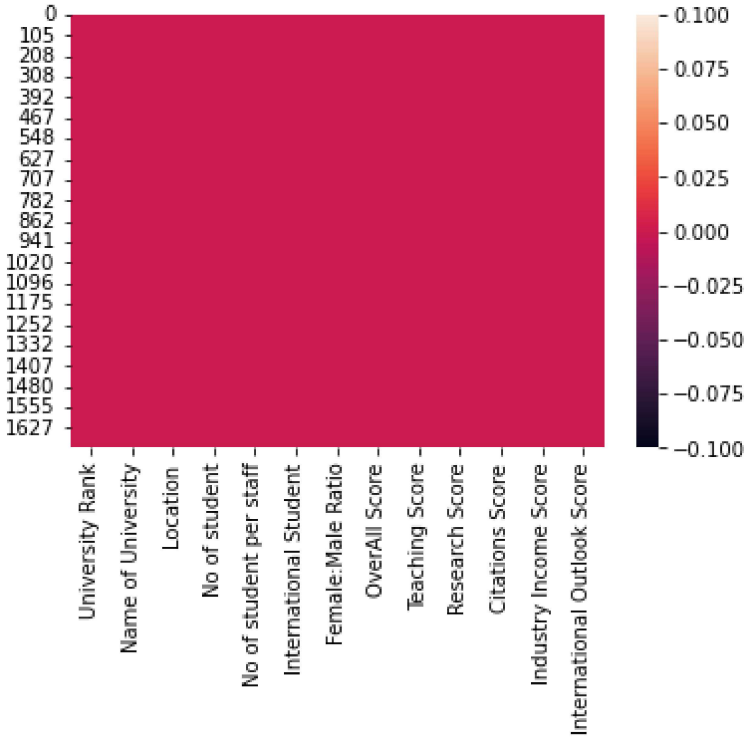


Drop Null Values

In [15]:
```
df.dropna(axis = 0, inplace =True)
```

In [16]:
```
sns.heatmap(df.isnull())
```

Out[16]: <AxesSubplot:>



Find Duplicate Values

In [17]:
```
df.duplicated().reset_index()
```

Out[17]:

	index	0
0	0	False
1	1	False
2	2	False
3	3	False
4	4	False
...	...	...
1483	1692	False
1484	1693	False
1485	1694	False
1486	1695	False
1487	1696	False

1488 rows × 2 columns

Heighest Teaching Score

In [18]:
```
df['Teaching Score'].max()
```

Out[18]: 94.8

Lowest Teaching Score

In [19]:
```
df['Teaching Score'].min()
```

Out[19]: 11.6

Teaching Score == 86.0

In [20]:
```
df[df['Teaching Score'] == 86.0]
```

Out[20]:

	University Rank	Name of University	Location	No of student	No of student per staff	International Student	Female:Male Ratio	OverAll Score	Teachin Scor
13	14	University of Pennsylvania	United States	21,453	6.3	23%	53 : 47	88.8	86.

Seperating female and male ratio columns

In [21]:
```
# Seperating female and male ratio columns
df[['Female Ratio', 'Male Ratio']] = df['Female:Male Ratio'].str.split(' : ', exp

# Converting the split columns to float
df['Female Ratio'] = df['Female Ratio'].astype(float)
df['Male Ratio'] = df['Male Ratio'].astype(float)

# Drop Original Column
df.drop('Female:Male Ratio', axis=1, inplace=True)
df.head()
```

Out[21]:

	University Rank	Name of University	Location	No of student	No of student per staff	International Student	OverAll Score	Teaching Score	Research Score
0	1	University of Oxford	United Kingdom	20,965	10.6	42%	96.4	92.3	99.7
1	2	Harvard University	United States	21,887	9.6	25%	95.2	94.8	99.0
2	3	University of Cambridge	United Kingdom	20,185	11.3	39%	94.8	90.9	99.5
3	3	Stanford University	United States	16,164	7.1	24%	94.8	94.2	96.7
4	5	Massachusetts Institute of Technology	United States	11,415	8.2	33%	94.2	90.7	93.6

Location wise International Students

In [42]:
```
df.groupby('Location')["International Student"].max().head().reset_index()
```

Out[42]:

	Location	International Student
0	Algeria	2%
1	Argentina	5%
2	Australia	51%
3	Austria	46%
4	Azerbaijan	2%

Top 5 Heighest Score Universities 2023

In [34]:
```
df.sort_values(by = 'OverAll Score', ascending = False).head()
```

Out[34]:

	University Rank	Name of University	Location	No of student	No of student per staff	International Student	OverAll Score	Teaching Score	Research Score
0	1	University of Oxford	United Kingdom	20,965	10.6	42%	96.4	92.3	99.7
1	2	Harvard University	United States	21,887	9.6	25%	95.2	94.8	99.0
2	3	University of Cambridge	United Kingdom	20,185	11.3	39%	94.8	90.9	99.5
3	3	Stanford University	United States	16,164	7.1	24%	94.8	94.2	96.7
4	5	Massachusetts Institute of Technology	United States	11,415	8.2	33%	94.2	90.7	93.6

In [46]:
```
df.groupby('Location').max().head()
```

Out[46]:

Location	University Rank	Name of University	No of student	No of student per staff	International Student	OverAll Score	Teaching Score	Research Score	Cita S
Algeria	401–500	Université Mouloud Mammeri de Tizi-Ouzou	41,841	24.4	2%	42.1–44.9	30.4	19.8	
Argentina	1501+	Universidad Nacional del Litoral	47,336	30.0	5%	18.4–24.3	18.3	10.4	
Australia	62	Western Sydney University	7,246	69.0	51%	77.6	67.1	75.9	
Austria	601–800	University of Vienna	7,274	42.9	46%	60.4	49.4	59.3	
Azerbaijan	1501+	Baku State University	23,323	13.3	2%	10.4–18.3	16.5	7.8	

What is the average number of students per staff member across all universities?

In [49]:
```
average_students_per_staff = df['No of student per staff'].mean()
average_students_per_staff
```

Out[49]: 18.83319892473117

What is the highest overall score in 2023?

In [51]:
```
highest_score_university = df['OverAll Score'].max()
highest_score_university
```

Out[51]: '96.4'

What is the highest Male Ration in 2023?

In [53]:
```
Male_rati = df['Male Ratio'].max()
Male_rati
```

Out[53]: 98.0

In [57]:
```
plt.figure(figsize=(30, 26))
sns.histplot(df['International Student'], bins=20, kde=True, color='skyblue')

plt.xlabel('International Student')
plt.ylabel('University Rank')
plt.title('Distribution of University Rank in 2023')

plt.show()
```