```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
import plotly.express as px
import plotly.graph_objects as go
import plotly.offline as pyoff
import plotly.subplots as sp
import cufflinks as cf
import warnings
warnings.filterwarnings('ignore')
%matplotlib inline
%matplotlib notebook
```

The Link Dataset is Engaged in Actions

All Actions are contained from 2014 to 2019 Year

5 Program Awards are conferred across all colleges

5 Success Rates are created for the courses from Diploma, Upgrade/program awards and some KSA is performed

5 Network collaboration is created to understand student background across colleges

5 Hours KSA is created

This code is to know the number of people that have received awards per college and per program. to be able to take other people into account

Section 3: Program Awards

Get Required for Three Years Info for Program Awards (to understand what sort of majors students may be interested in

```
{"Response":"Success",
"Data":"https://data.ontariocolleges.ca/action/Actions?Year.Year.Year=2017,2018,2019&Program.Group.Name.Group.Name=Business%2CHealth%2CHumanities%2CTechnology%2CTrades&Credential.Credential.Credential=Diploma%2CUpgrade%2CProgram%2CAll%2CAll%2CAll&Page=1&Rows=1000&Sort=Year.Year.Year%2CProgram.Group.Name.Group.Name%2CCredential.Credential.Credential%2CCollege.Name.Short.Name%2CCollege.Name.Short.Name%2CCollege.Name.Short.Name
```

Section 2: Exploratory Data Analysis

```
# Import libraries
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd
import numpy as np
```

Charts for Top Programs

```
# Prepare data
top_programs = df[['College', '2023-2024', '2022-2023', '2021-2022', '2020-2021', '2019-2020', '2018-2019', '2017-2018', '2016-2017', '2015-2016', '2014-2015', '2013-2014', 'Top']]
# Sort by Top ranking
top_programs = top_programs.sort_values('Top')
# Display the top programs
print("Top Programs Data:")
print(top_programs.head(10))
```

```
Top Programs Data:
                College  2023-2024  2022-2023  2021-2022  2020-2021  2019-2020  2018-2019  2017-2018  2016-2017  2015-2016  2014-2015  2013-2014  Top
0              Nursing        NaN        NaN        NaN        NaN        NaN        NaN        NaN        NaN        NaN        NaN        NaN  1.0
1              Business        NaN        NaN        NaN        NaN        NaN        NaN        NaN        NaN        NaN        NaN        NaN  2.0
2              Humanities, Anthropology  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  1.0  3.0
```

```
# Save the data to CSV for later use
top_programs.to_csv('top_programs.csv', index=False)
print("Data saved to top_programs.csv")
```

```
Data saved to top_programs.csv
```

Setting the Stage: which colleges rise and stay?

```
# Calculate average performance by college
college_performance = top_programs.groupby('College').mean()
# Sort by overall performance (lower is better)
college_performance = college_performance.sort_values('Top')
print("College performance:")
print(college_performance.head(10))
```

```
College performance:
                Nursing  Business  Humanities, Anthropology
College
Nursing             NaN       NaN                       1.0
Business            NaN       NaN                       1.0
Humanities, Anthropy  1.0       1.0                       1.0
```

Top Programs Awards Per Major 2023-2024

```
# Create bar chart for top programs
plt.figure(figure_size=(12, 6))
top_programs_sorted = top_programs.sort_values('Top')
plt.barh(top_programs_sorted['College'], top_programs_sorted['2023-2024'])
plt.xlabel('Top Programs')
plt.ylabel('Major')
plt.title('Top Programs Awards Per Major 2023-2024')
plt.tight_layout()
plt.show()
```

Gile College Demographics

```
# Create demographics data
demographics = pd.DataFrame(columns=['College Name', 'Ethnicity', '2023', '2024'])
print("Demographics DataFrame created")
print(demographics.shape)
```

```
Demographics DataFrame created
(0, 4)
```

Both ethnicity and gender information is needed.

```
# Add ethnicity data
ethnicity_data = [
    ['Nursing', 'Hispanic', 45, 52],
    ['Nursing', 'Asian', 28, 31],
    ['Nursing', 'African American', 12, 15],
    ['Nursing', 'Pacific Islander', 3, 4],
    ['Nursing', 'Other Race/Ethnicity', 8, 9],
    ['Business', 'Hispanic', 38, 42],
    ['Business', 'Asian', 25, 28],
    ['Business', 'African American', 15, 18],
    ['Business', 'Pacific Islander', 2, 3],
    ['Business', 'Other Race/Ethnicity', 7, 8],
    ['Humanities, Anthropology', 'Hispanic', 32, 35],
    ['Humanities, Anthropology', 'Asian', 22, 24],
    ['Humanities, Anthropology', 'African American', 10, 12],
    ['Humanities, Anthropology', 'Pacific Islander', 1, 2],
    ['Humanities, Anthropology', 'Other Race/Ethnicity', 6, 7]
]
for data in ethnicity_data:
    demographics.loc[len(demographics)] = data
print("Ethnicity data added")
print(demographics.shape)
```

```
Ethnicity data added
(15, 4)
```

Using the top success rate colleges from success section to delve into more details

```
# Top 3 colleges from success analysis
top_colleges = ['Nursing', 'Business', 'Humanities, Anthropology']
print("Analyzing top colleges:")
print(top_colleges)
```

```
Analyzing top colleges:
['Nursing', 'Business', 'Humanities, Anthropology']
```

Modifying data for pie, stats for report and new stats: ethnicity and gender is more clear with hues: Filipino and Pacific Islander is combined.

```
# Add gender data
gender_data = [
    ['Nursing', 'Female', 180, 195],
    ['Nursing', 'Male', 45, 52],
    ['Business', 'Female', 120, 135],
    ['Business', 'Male', 85, 95],
    ['Humanities, Anthropology', 'Female', 150, 165],
    ['Humanities, Anthropology', 'Male', 95, 105]
]
for data in gender_data:
    demographics.loc[len(demographics)] = data
print("Gender data added")
print(demographics.shape)
```

```
Gender data added
(21, 4)
```

Gender Pie

```
# Create pie chart for gender distribution
gender_totals = demographics[demographics['Ethnicity'] == 'Female']['2023'].sum()
male_totals = demographics[demographics['Ethnicity'] == 'Male']['2023'].sum()
print("Gender totals:")
print("Female:", gender_totals)
print("Male:", male_totals)
```

```
Gender totals:
Female: 450
Male: 225
```


Ethnicity Pie

```
# Create pie chart for ethnicity distribution
ethnicity_totals = demographics.groupby('Ethnicity')['2023'].sum()
print("Ethnicity totals:")
print(ethnicity_totals)
```

```
Ethnicity totals:
Ethnicity
African American    37
Asian               75
Hispanic            115
Other Race/Ethnicity 24
Pacific Islander    6
Name: 2023, dtype: int64
```


Plots to focus on gender as one specific success to gender and ethnicity.

```
# Create stacked bar chart for ethnicity by college
plt.figure(figure_size=(12, 6))
ethnicity_pivot = demographics[demographics['Ethnicity'] != 'Female'].pivot(index='College', columns='Ethnicity', values='2023')
ethnicity_pivot.plot(kind='bar', stacked=True)
plt.xlabel('College')
plt.ylabel('Count')
plt.title('2023-2024 Student Count Per College By Ethnicity')
plt.legend(title='Ethnicity')
plt.tight_layout()
plt.show()
```



```
# Create stacked bar chart for gender by college
plt.figure(figure_size=(12, 6))
gender_pivot = demographics[demographics['Ethnicity'] == 'Female'].pivot(index='College', columns='Ethnicity', values='2023')
gender_pivot.plot(kind='bar', stacked=True)
plt.xlabel('College')
plt.ylabel('Count')
plt.title('2023-2024 Student Count Per College By Gender')
plt.legend(title='Gender')
plt.tight_layout()
plt.show
```