# DATA ANALYST

## DATASET: UDEMY COURSES

PREPARED BY: JOE MANDE

TOOL USED: PYTHON FOR DATA ANALYSIS

## TABLE OF CONTENTS



## 1. IMPORT PYTHON LIBRARIES

Numpy Libraries

- For data consolidation.

Pandas Libraries

- To load the dataset.
- For data cleaning and data analysis.

Matplotlib Libraries

- For data visualization.

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

## 2. LOADING THE DATASET

Four datasets were loaded for each course:

- Business Finance
- Music Instruments
- Web Development
- Graphic Design

In [2]:
```
business_finance = pd.read_csv('/Users/joemande/Downloads/business.csv')
music_instruments = pd.read_csv('/Users/joemande/Downloads/music.csv')
web_development = pd.read_csv('/Users/joemande/Downloads/web.csv')
graphic_design = pd.read_csv('/Users/joemande/Downloads/design.csv')
```

## 3. DATA CONSOLIDATION

The four datasets were consolidated to make one dataset.

In [3]:
```
data_consolidation = np.vstack([business_finance, music_instruments, web_development, gr
data = pd.DataFrame(data_consolidation)
data.head()
```

Out[3]:

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 49798.0 | Bitcoin or How I Learned to Stop Worrying and ... | https://www.udemy.com/bitcoin-or-how-i-learned... | 0.0 | 65576.0 | 936.0 | 24.0 | All Levels | 0.56 |
| **1** | 48841.0 | Accounting in 60 Minutes - A Brief Introduction | https://www.udemy.com/accounting-in-60-minutes... | 0.0 | 56659.0 | 4397.0 | 16.0 | Beginner Level | 0.95 |
| **2** | 133536.0 | Stock Market Investing for Beginners | https://www.udemy.com/the-beginners-guide-to-t... | 0.0 | 50855.0 | 2698.0 | 15.0 | All Levels | 0.91 |
| **3** | 151668.0 | Introduction to Financial Modeling | https://www.udemy.com/financial-modeling-asimp... | 0.0 | 29167.0 | 1463.0 | 8.0 | All Levels | 0.18 |
| **4** | 648826.0 | The Complete Financial Analyst Course 2017 | https://www.udemy.com/the-complete-financial-a... | 195.0 | 24481.0 | 2347.0 | 174.0 | All Levels | 0.37 |

## 4. DATA CLEANING

1. To ensure to have clear and concise names for headers

In [4]:
```
data.columns=['course_id', 'course_title', 'url', 'price', 'num_subscribers', 'num_revie
              'level','Rating','content_duration','published_timestamp','subject']
data.head()
```

Out[4]:

| | course_id | course_title | url | price | num_subscribers | num_reviews | num_ |
|---|---|---|---|---|---|---|---|
| **0** | 49798.0 | Bitcoin or How I Learned to Stop Worrying and ... | https://www.udemy.com/bitcoin-or-how-i-learned... | 0.0 | 65576.0 | 936.0 | |
| **1** | 48841.0 | Accounting in 60 Minutes - A Brief Introduction | https://www.udemy.com/accounting-in-60-minutes... | 0.0 | 56659.0 | 4397.0 | |
| **2** | 133536.0 | Stock Market Investing for Beginners | https://www.udemy.com/the-beginners-guide-to-t... | 0.0 | 50855.0 | 2698.0 | |
| **3** | 151668.0 | Introduction to Financial Modeling | https://www.udemy.com/financial-modeling-asimp... | 0.0 | 29167.0 | 1463.0 | |

| | | | | | | |
|---|---|---|---|---|---|---|
| **4** | 648826.0 | The Complete Financial Analyst Course 2017 | https://www.udemy.com/the-complete-financial-a... | 195.0 | 24481.0 | 2347.0 |

1. To delete any blank cells

In [5]:
```
data = data.dropna()
```

1. To remove any duplicates

In [6]:
```
data = data.drop_duplicates()
```

1. To ensure that the data is consistent so that we can easily understand what each column represents

In [7]:
```
data['subject'] = data['subject'].str.replace('Subject: Web Development',
                                              'Web Development')
```

## 4. DATA ANALYSIS AND VISUALIZATION

1. The total number of subscribers for each subject

- Table

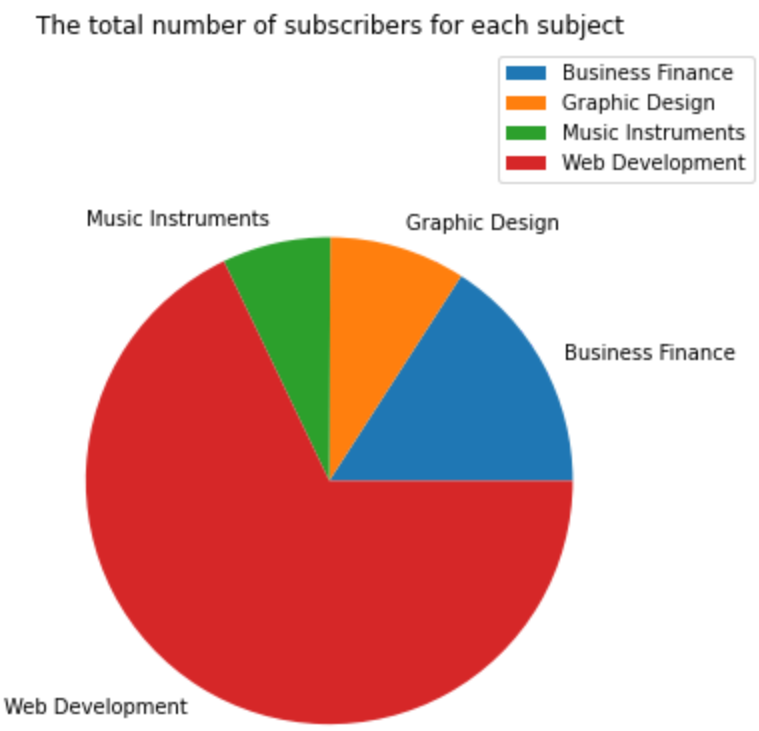
In [8]:
```
first_analysis = data.groupby(['subject'])['num_subscribers'].sum()
pd.DataFrame(first_analysis)
```

Out[8]:

| subject | num_subscribers |
|---|---|
| **Business Finance** | 1868711.0 |
| **Graphic Design** | 1063148.0 |
| **Musical Instruments** | 846689 |
| **Web Development** | 7981935.0 |

- Chart

In [9]:
```
plt.subplots(figsize=(8,8))
plt.title('The total number of subscribers for each subject')
plt.pie(first_analysis, radius=0.7, labels=['Business Finance', 'Graphic Design',
                                            'Music Instruments', 'Web Development'])
plt.legend()
plt.show()
```




1. The average number of subscribers for each subject

- Table

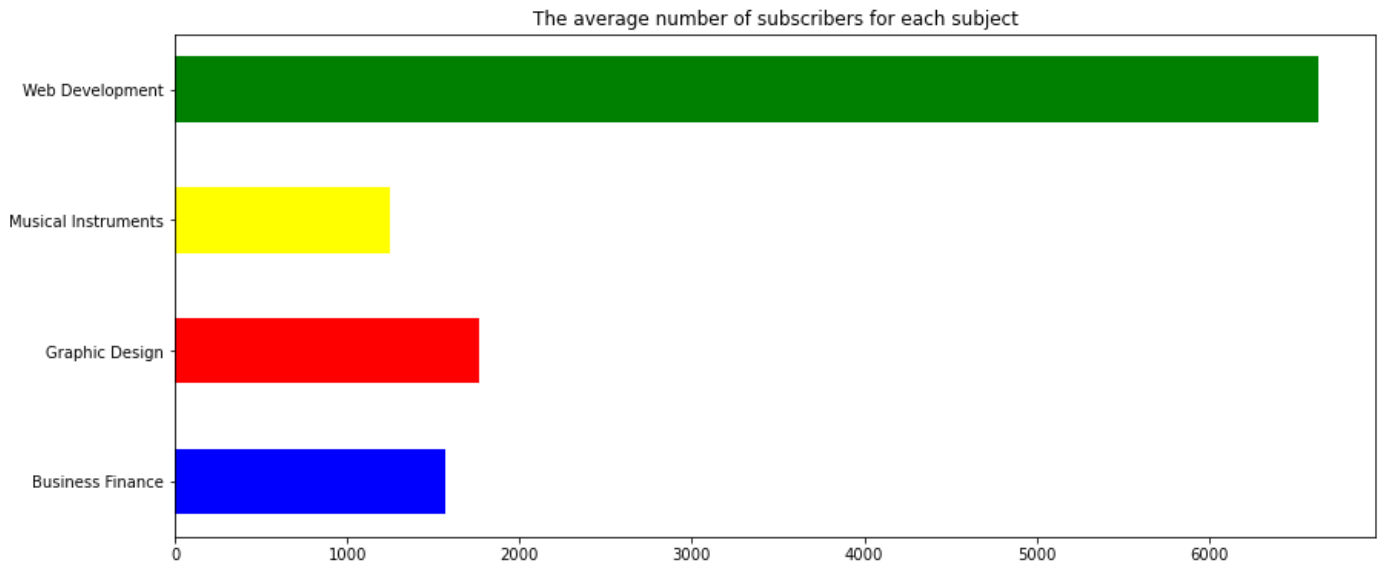
In [10]:
```
second_analysis = data.groupby(['subject'])['num_subscribers'].mean().round(2)
pd.DataFrame(second_analysis)
```

Out[10]:

| subject | num_subscribers |
|---|---|
| **Business Finance** | 1569.03 |
| **Graphic Design** | 1766.03 |
| **Musical Instruments** | 1245.13 |
| **Web Development** | 6635.02 |

- Chart

In [11]:
```
plt.subplots(figsize = (14,6))
plt.title('The average number of subscribers for each subject')
plt.barh(np.arange(len(second_analysis)), second_analysis, height=0.5,
        color=['blue', 'red', 'yellow', 'green'],
        tick_label=['Business Finance', 'Graphic Design',
                    'Musical Instruments', 'Web Development'])
plt.show()
```




1. The average cost per subject at each level

- Table

In [12]:
```
third_analysis = data.groupby(['subject','level'])['price'].mean().round(2)
pd.DataFrame(third_analysis)
```

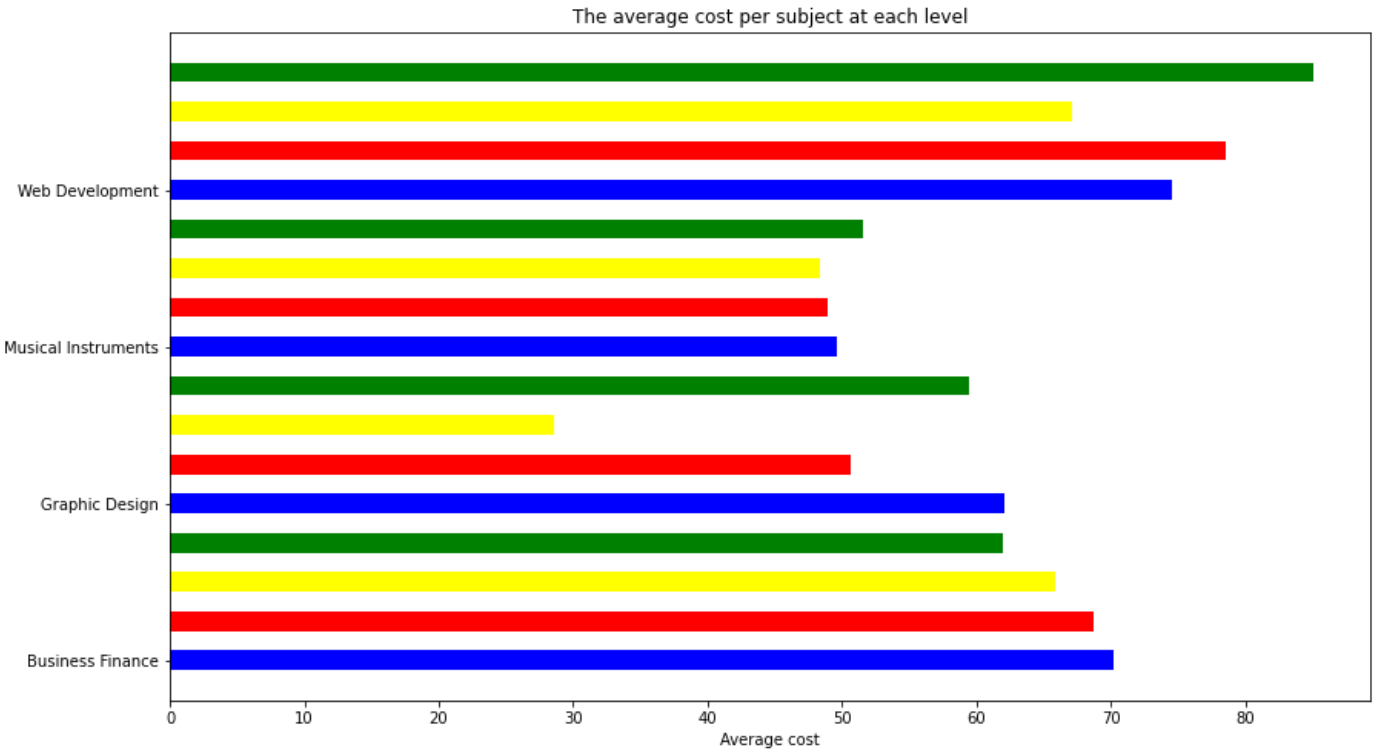
Out[12]:

| | | price |
|---|---|---|
| **subject** | **level** | |
| **Business Finance** | **All Levels** | 70.20 |
| | **Beginner Level** | 68.73 |
| | **Expert Level** | 65.80 |
| | **Intermediate Level** | 62.01 |
| **Graphic Design** | **All Levels** | 62.12 |
| | **Beginner Level** | 50.68 |
| | **Expert Level** | 28.57 |
| | **Intermediate Level** | 59.41 |
| **Musical Instruments** | **All Levels** | 49.58 |
| | **Beginner Level** | 48.98 |
| | **Expert Level** | 48.33 |
| | **Intermediate Level** | 51.60 |
| **Web Development** | **All Levels** | 74.55 |
| | **Beginner Level** | 78.54 |
| | **Expert Level** | 67.14 |
| | **Intermediate Level** | 85.07 |

- Chart

In [13]:
```
plt.subplots(figsize=(14,8))
plt.title('The average cost per subject at each level')
plt.barh(np.arange(len(third_analysis)), third_analysis, height=0.5,
```

```
        color=['blue', 'red', 'yellow', 'green'])
plt.xlabel('Average cost')
plt.yticks([0, 4, 8, 12], labels=['Business Finance', 'Graphic Design',
                                  'Musical Instruments', 'Web Development'])
plt.show()
```




1. The average content duration for each subject

- Table

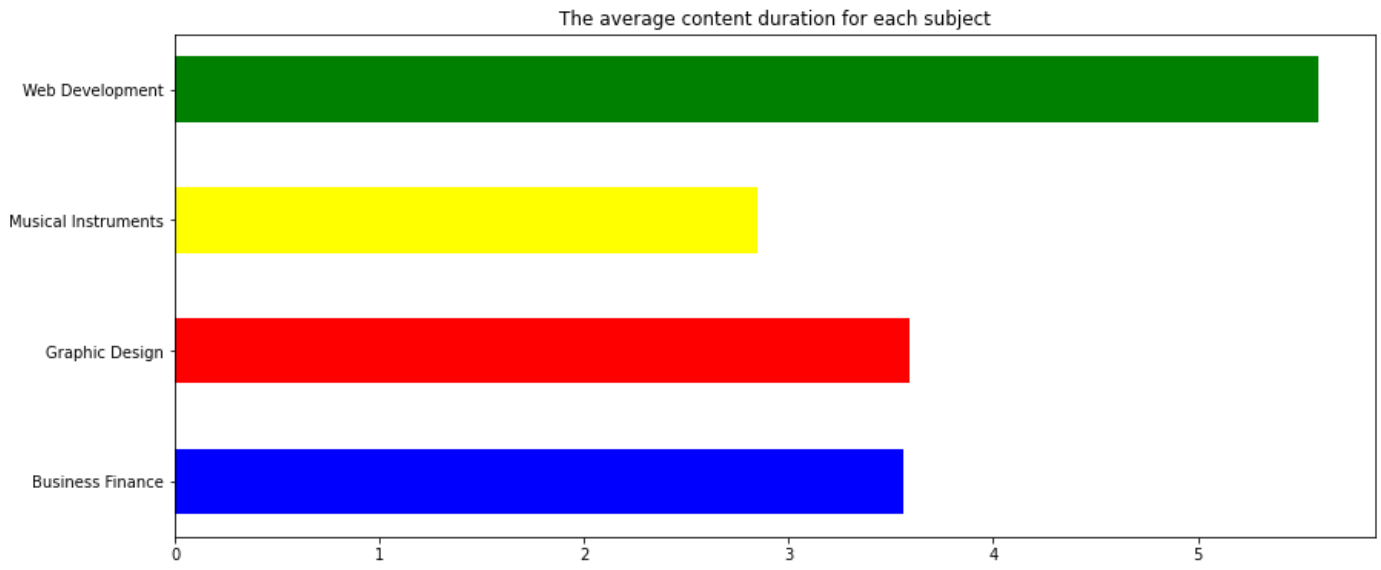
In [14]:
```
fourth_analysis = data.groupby(['subject'])['content_duration'].mean().round(2)
pd.DataFrame(fourth_analysis)
```

Out[14]:

| subject | content_duration |
|---|---|
| **Business Finance** | 3.56 |
| **Graphic Design** | 3.59 |
| **Musical Instruments** | 2.85 |
| **Web Development** | 5.59 |

- Chart

In [15]:
```
plt.subplots(figsize=(14,6))
plt.title('The average content duration for each subject')
plt.barh(np.arange(len(fourth_analysis)), fourth_analysis, height=0.5,
        color=['blue', 'red', 'yellow', 'green'],
        tick_label=['Business Finance', 'Graphic Design',
                    'Musical Instruments', 'Web Development'])
plt.show()
```




1. The average rating per subject at each level

- Table

In [16]:
```
fifth_analysis = data.groupby(['subject', 'level'])['Rating'].mean().round(2)
pd.DataFrame(fifth_analysis)
```

Out[16]:

| | | Rating |
|---|---|---|
| **subject** | **level** | |
| **Business Finance** | **All Levels** | 0.69 |
| | **Beginner Level** | 0.69 |
| | **Expert Level** | 0.70 |
| | **Intermediate Level** | 0.70 |
| **Graphic Design** | **All Levels** | 0.73 |
| | **Beginner Level** | 0.73 |
| | **Expert Level** | 0.88 |
| | **Intermediate Level** | 0.72 |
| **Musical Instruments** | **All Levels** | 0.31 |
| | **Beginner Level** | 0.31 |
| | **Expert Level** | 0.30 |
| | **Intermediate Level** | 0.28 |
| **Web Development** | **All Levels** | 0.65 |
| | **Beginner Level** | 0.64 |
| | **Expert Level** | 0.50 |
| | **Intermediate Level** | 0.67 |

- Chart

In [17]:
```
plt.subplots(figsize=(14,6))
plt.title('The average rating per subject at each level')
plt.bar(np.arange(len(fifth_analysis)), fifth_analysis, width=0.5,
```

```
        color=['blue', 'red', 'yellow', 'green'])
plt.ylabel('Average rating')
plt.xticks([0, 4, 8, 12], labels=['Business Finance', 'Graphic Design',
                                  'Musical Instruments', 'Web Development'])
plt.xticks(rotation=90)
plt.show()
```