

Data Analysis for Learning Experience with AWS - A Pilot Study

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

```
In [3]: # Import data and assign to df (pandas DataFrame)
data_path = "Final_Reflections_Data.csv"
df = pd.read_csv(data_path)
```

```
In [4]: # Print number of rows and columns
rows, columns = df.shape
print(f"Number of rows: {rows}")
print(f"Number of columns: {columns}")
```

Number of rows: 24
Number of columns: 27

```
In [5]: # Check the names of all columns in the df
print(df.columns)
```

```
Index(['DateTime', 'Email', 'Student_Name', 'StudentID', 'Date',
      'Course_Title', 'Course_Duration_Hours', 'Subject_Knowledge_Before',
      'Platform_Experience', 'Ease_of_Navigation',
      'Course_Content_Organisation', 'Usability_Of_Interactive_Elements',
      'Platform_Support_For_Learning',
      'Skills_Acquired_Relevant_To_Career_Aspiration', 'Course_Engaging',
      'Motivation_To_Learn_More', 'Challenges_Faced',
      'Confidence_In_Applying_Skills', 'Platform_Contribution_To_Learning',
      'Course_Recommendation_To_Others', 'Subject_Knowledge_After',
      'Understanding_Of_Topic_After', 'Assessment_Completed',
      'First_Attempt_Score', 'Number_Of_Attempts',
      'Hours_Taken_To_Complete_Course', 'First_Attempt_Score_No'],
      dtype='object')
```

```
In [6]: # Get a quick look at the data (display first 6 rows)
df.head(4)
```

```
Out[6]:
```

	DateTime	Email	Student_Name	StudentID	Date	Course_Title	Course_Duration_H
0	9/7/2024 15:15	faisalcox2015@gmail.com	Mohammad Faisal	u3212031	7/9/2024	Becoming a Cloud Practitioner - Part 1 - Cloud...	
1	9/8/2024 1:30	tw.tangbi@gmail.com	Tshering Wangchuk	u3254369	9/8/2024	Becoming a Cloud Practitioner - Part 1 - Cloud...	
2	9/8/2024 14:49	cringtoby07@gmail.com	Tshering Tobgyel	U3259355	8/9/2024	Becoming a Cloud Practitioner - Part 1 - Cloud...	
3	9/8/2024 15:05	cringtoby07@gmail.com	Tshering Tobgyel	U3259355	8/9/2024	Becoming a Cloud Practitioner - Part 2 - Compu...	

4 rows × 27 columns

```
In [7]: # Get a summary of df
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 24 entries, 0 to 23
Data columns (total 27 columns):
#      Column                                     Non-Null Count  Dtype
---  -
0     DateTime                                     24 non-null    object
1     Email                                       24 non-null    object
2     Student_Name                             24 non-null    object
3     StudentID                                24 non-null    object
4     Date                                       24 non-null    object
5     Course_Title                             24 non-null    object
6     Course_Duration_Hours                    24 non-null    float64
7     Subject_Knowledge_Before                 24 non-null    object
8     Platform_Experience                      24 non-null    int64
9     Ease_of_Navigation                       24 non-null    int64
10    Course_Content_Organisation              24 non-null    int64
11    Usability_Of_Interactive_Elements        24 non-null    int64
12    Platform_Support_For_Learning            24 non-null    object
13    Skills_Acquired_Relevant_To_Career_Aspiration 24 non-null    int64
14    Course_Engaging                         24 non-null    int64
15    Motivation_To_Learn_More                 24 non-null    int64
16    Challenges_Faced                         24 non-null    object
17    Confidence_In_Applying_Skills            24 non-null    int64
18    Platform_Contribution_To_Learning        24 non-null    object
19    Course_Recommendation_To_Others          24 non-null    object
20    Subject_Knowledge_After                  24 non-null    object
21    Understanding_Of_Topic_After             24 non-null    int64
22    Assessment_Completed                     24 non-null    object
23    First_Attempt_Score                      24 non-null    object
24    Number_Of_Attempts                       24 non-null    int64
25    Hours_Taken_To_Complete_Course           24 non-null    float64
26    First_Attempt_Score_No                   24 non-null    int64
dtypes: float64(2), int64(11), object(14)
memory usage: 5.2+ KB
```

Data Cleaning

```
In [9]: # Convert the column to lowercase and replace 'yes' with 1 and 'no' with 0
df['Platform_Support_For_Learning'] = df['Platform_Support_For_Learning'].str.lower().replace({'
```

```
/var/folders/1d/z6pffxjs2l9cm5q8fjh804580000gn/T/ipykernel_21885/1285141336.py:2: FutureWarning:
Downcasting behavior in `replace` is deprecated and will be removed in a future version. To retain
the old behavior, explicitly call `result.infer_objects(copy=False)`. To opt-in to the future
behavior, set `pd.set_option('future.no_silent_downcasting', True)`
  df['Platform_Support_For_Learning'] = df['Platform_Support_For_Learning'].str.lower().replace
  ({'yes': 1, 'no': 0}).astype(int)
```

```
In [10]: # Replace 'Yes' with 1 and 'No' with 0, then cast the column to integer
df['Assessment_Completed'] = df['Assessment_Completed'].replace({'Yes': 1, 'No': 0}).astype(int)
```

```
/var/folders/1d/z6pffxjs2l9cm5q8fjh804580000gn/T/ipykernel_21885/3004353526.py:2: FutureWarning:
Downcasting behavior in `replace` is deprecated and will be removed in a future version. To retain
the old behavior, explicitly call `result.infer_objects(copy=False)`. To opt-in to the future
behavior, set `pd.set_option('future.no_silent_downcasting', True)`
  df['Assessment_Completed'] = df['Assessment_Completed'].replace({'Yes': 1, 'No': 0}).astype(int)
```

```
In [11]: df.head(23)
```

Out[11]:

	DateTime	Email	Student_Name	StudentID	Date	Course_Title	Course_Duration
0	9/7/2024 15:15	faisalcox2015@gmail.com	Mohammad Faisal	u3212031	7/9/2024	Becoming a Cloud Practitioner - Part 1 - Cloud...	
1	9/8/2024 1:30	tw.tangbi@gmail.com	Tshering Wangchuk	u3254369	9/8/2024	Becoming a Cloud Practitioner - Part 1 - Cloud...	
2	9/8/2024 14:49	cringtoby07@gmail.com	Tshering Tobgyel	U3259355	8/9/2024	Becoming a Cloud Practitioner - Part 1 - Cloud...	
3	9/8/2024 15:05	cringtoby07@gmail.com	Tshering Tobgyel	U3259355	8/9/2024	Becoming a Cloud Practitioner - Part 2 - Compu...	
4	9/9/2024 0:55	tw.tangbi@gmail.com	Tshering Wangchuk	u3254369	9/9/2024	Becoming a Cloud Practitioner - Part 2 - Compu...	
5	9/9/2024 23:17	faisalcox2015@gmail.com	Mohammad Faisal	u3212031	9/9/2024	Becoming a Cloud Practitioner - Part 2 - Compu...	
6	9/10/2024 10:38	noofz2017@gmail.com	Nouf	U3226243	9/10/2024	Becoming a Cloud Practitioner - Part 1 - Cloud...	
7	9/14/2024 15:50:48	tw.tangbi@gmail.com	Tshering Wangchuk	u3254369	9/14/2024	Becoming a Cloud Practitioner - Part 3 - Ident...	
8	9/14/2024 17:11:12	faisalcox2015@gmail.com	Mohammad Faisal	u3212031	9/14/2024	Becoming a Cloud Practitioner - Part 3 - Ident...	
9	9/15/2024 16:15:49	tw.tangbi@gmail.com	Tshering Wangchuk	u3254369	9/15/2024	Becoming a Cloud Practitioner - Part 4 - Advan...	
10	9/15/2024 21:32:00	faisalcox2015@gmail.com	Mohammad Faisal	u3212031	9/15/2024	Becoming a Cloud Practitioner - Part 4 - Advan...	
11	9/15/2024 23:24:48	noofz2017@gmail.com	Nouf	U3226243	9/9/2024	Becoming a Cloud Practitioner - Part 2 - Compu...	
12	9/15/2024 23:38:30	noofz2017@gmail.com	Nouf	U3226243	9/14/2024	Becoming a Cloud Practitioner - Part 3 - Ident...	
13	9/15/2024 23:47:33	noofz2017@gmail.com	Nouf	U3226243	9/14/2024	Becoming a Cloud	

	DateTime	Email	Student_Name	StudentID	Date	Course_Title	Course_Duration
						Practitioner - Part 4 - Advan...	
14	9/16/2024 14:28:00	cringtoby07@gmail.com	Tshering Tobgyel	U3259355	9/16/2024	Becoming a Cloud Practitioner - Part 4 - Advan...	
15	9/16/2024 22:07:36	cringtoby07@gmail.com	Tshering Tobgyel	U3259355	9/16/2024	Becoming a Cloud Practitioner - Part 3 - Ident...	
16	9/25/2024 14:40:02	tw.tangbi@gmail.com	Tshering Wangchuk	u3254369	9/25/2024	AWS Cloud Practitioner Essentials	
17	9/26/2024 14:17:41	cringtoby07@gmail.com	Tshering Tobgyel	U3259355	9/26/2024	AWS Technical Essentials	
18	10/1/2024 21:13	faisalcox2015@gmail.com	Mohammad Faisal	u3212031	10/1/2024	AWS Cloud Practitioner Essentials	
19	10/2/2024 0:08	faisalcox2015@gmail.com	Mohammad Faisal	u3212031	10/9/2024	AWS Technical Essentials	
20	10/2/2024 18:39	noofz2017@gmail.com	Nouf	U3226243	9/30/2024	AWS Technical Essentials	
21	10/2/2024 18:50	noofz2017@gmail.com	Nouf	U3226243	9/30/2024	AWS Cloud Practitioner Essentials	
22	10/3/2024 0:45	tw.tangbi@gmail.com	Tshering Wangchuk	u3254369	10/3/2024	AWS Technical Essentials	

23 rows x 27 columns

```
In [12]: # Get Unique Courses
unique_course_titles = df['Course_Title'].unique()
print("Number of courses taken: ", len(unique_course_titles))
print("Unique Courses:")
print(unique_course_titles)
```

Number of courses taken: 6

Unique Courses:

```
['Becoming a Cloud Practitioner - Part 1 - Cloud Basics'
 'Becoming a Cloud Practitioner - Part 2 - Compute, Networking, and Account Strategies'
 'Becoming a Cloud Practitioner - Part 3 - Identities, Security, and Monitoring the AWS Cloud'
 'Becoming a Cloud Practitioner - Part 4 - Advanced Cloud Services'
 'AWS Cloud Practitioner Essentials' 'AWS Technical Essentials']
```

```
In [13]: print("Course taken by number of people: ")
df['Course_Title'].value_counts()
```

Course taken by number of people:

```
Out[13]: Course_Title
Becoming a Cloud Practitioner – Part 1 – Cloud Basics
4
Becoming a Cloud Practitioner – Part 2 – Compute, Networking, and Account Strategies
4
Becoming a Cloud Practitioner – Part 3 – Identities, Security, and Monitoring the AWS Cloud
4
Becoming a Cloud Practitioner – Part 4 – Advanced Cloud Services
4
AWS Cloud Practitioner Essentials
4
AWS Technical Essentials
4
Name: count, dtype: int64
```

```
In [14]: print("Course Duration Descriptive Statistics:")
df['Course_Duration_Hours'].describe()
```

Course Duration Descriptive Statistics:

```
Out[14]: count    24.00000
mean      3.95000
std       1.50795
min       2.45000
25%      2.75000
50%      3.75000
75%      4.00000
max       7.00000
Name: Course_Duration_Hours, dtype: float64
```

```
In [15]: print("Time Taken to complete the course Descriptive Statistics:")
df['Hours_Taken_To_Complete_Course'].describe()
```

Time Taken to complete the course Descriptive Statistics:

```
Out[15]: count    24.000000
mean      4.614583
std       2.427753
min       2.000000
25%      2.882500
50%      4.050000
75%      6.017500
max      10.280000
Name: Hours_Taken_To_Complete_Course, dtype: float64
```

```
In [16]: # Count the occurrences of each category
knowledge_counts = df['Subject_Knowledge_Before'].value_counts()

knowledge_counts
```

```
Out[16]: Subject_Knowledge_Before
Beginner      12
No Knowledge At All    9
Intermediate    3
Name: count, dtype: int64
```

```
In [17]: # Define a color palette
colors = plt.get_cmap('Accent')

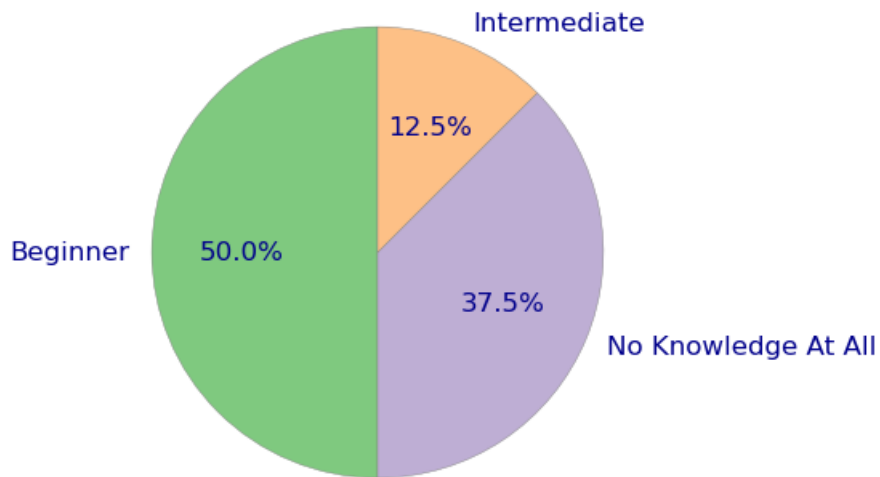
# Plot pie chart
plt.figure(figsize=(4, 4))
plt.pie(knowledge_counts, labels=knowledge_counts.index, autopct='%1.1f%%', startangle=90,
        colors=colors(range(len(knowledge_counts))),
        wedgeprops={'linewidth':0.3, 'edgecolor': 'grey'})

plt.title('Subject Knowledge Before Beginning Course', fontsize=14)
plt.axis('equal') # Equal aspect ratio ensures that pie is drawn as a circle.

# Enhance the text properties
for text in plt.gca().texts:
    text.set_fontsize(12) # Increase text size
    text.set_color('darkblue') # Change text color

plt.show()
```

Subject Knowledge Before Beginning Course



```
In [18]: # Count the occurrences of each category
knowledge_counts_after = df['Subject_Knowledge_After'].value_counts()

knowledge_counts_after
```

```
Out[18]: Subject_Knowledge_After
Intermediate    20
Beginner        4
Name: count, dtype: int64
```

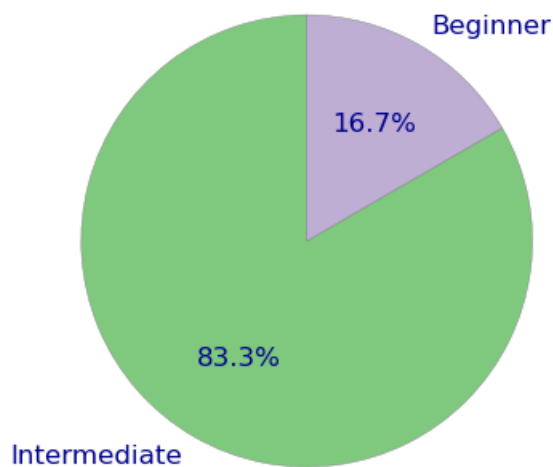
```
In [19]: # Plot pie chart
plt.figure(figsize=(4, 4))
plt.pie(knowledge_counts_after, labels=knowledge_counts_after.index, autopct='%1.1f%%', startangle=90,
        colors=colors(range(len(knowledge_counts_after))),
        wedgeprops={'linewidth':0.3, 'edgecolor': 'grey'})

plt.title('Subject Knowledge After Completing the Course', fontsize=14)
plt.axis('equal')

# Enhance the text properties
for text in plt.gca().texts:
    text.set_fontsize(12)
    text.set_color('darkblue')

plt.show()
```

Subject Knowledge After Completing the Course



```
In [20]: # Mapping from integer to descriptive string
understanding_mapping = {
    1: 'No Improvement',
    2: 'Minimal Improvement',
    3: 'Moderate Improvement',
    4: 'Significant Improvement',
```

```

5: 'Comprehensive Improvement'
}

# Apply mapping
df['Understanding_Of_Topic_After_Label'] = df['Understanding_Of_Topic_After'].map(understanding_

```

```

In [21]: # Count occurrences
Understanding_Of_Topic_After_Label_Counts = df['Understanding_Of_Topic_After_Label'].value_counts()

Understanding_Of_Topic_After_Label_Counts

```

```

Out[21]: Understanding_Of_Topic_After_Label
No Improvement      0
Minimal Improvement  0
Moderate Improvement  2
Significant Improvement  19
Comprehensive Improvement  3
Name: count, dtype: int64

```

```

In [22]: # Set the plot size
plt.figure(figsize=(7, 4))

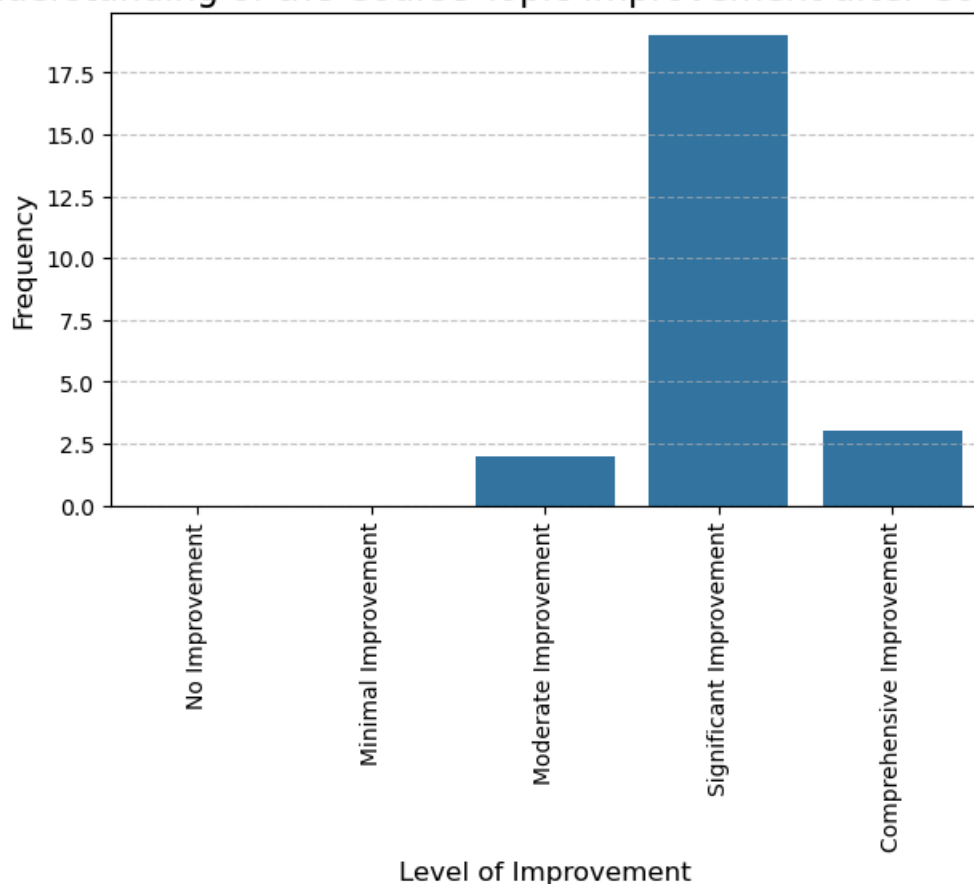
# Use Seaborn to create a countplot
sns.countplot(data=df, x='Understanding_Of_Topic_After_Label', order=understanding_mapping.values())

# Enhance plot appearance
plt.title('Understanding of the Course Topic Improvement after Completion', fontsize=16)
plt.xlabel('Level of Improvement', fontsize=12)
plt.ylabel('Frequency', fontsize=12)
plt.xticks(rotation=90)
plt.grid(axis='y', linestyle='--', alpha=0.7)

# Show the plot
plt.show()

```

Understanding of the Course Topic Improvement after Completion



```

In [23]: # Define the mapping for both columns
platform_mapping = {
    1: 'Very Difficult',
    2: 'Somewhat Difficult',

```

```

3: 'Neutral',
4: 'Easy',
5: 'Very Easy'
}

```

```

In [24]: # Apply mapping to both columns
df['Platform_Experience_Label'] = df['Platform_Experience'].map(platform_mapping)
df['Ease_of_Navigation_Label'] = df['Ease_of_Navigation'].map(platform_mapping)

```

```

In [25]: df.info()

```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 24 entries, 0 to 23
Data columns (total 30 columns):
#   Column                                                                 Non-Null Count  Dtype
---  -
0   DateTime                                                                24 non-null    object
1   Email                                                                    24 non-null    object
2   Student_Name                                                            24 non-null    object
3   StudentID                                                                24 non-null    object
4   Date                                                                    24 non-null    object
5   Course_Title                                                            24 non-null    object
6   Course_Duration_Hours                                                  24 non-null    float64
7   Subject_Knowledge_Before                                                24 non-null    object
8   Platform_Experience                                                     24 non-null    int64
9   Ease_of_Navigation                                                      24 non-null    int64
10  Course_Content_Organisation                                              24 non-null    int64
11  Usability_Of_Interactive_Elements                                       24 non-null    int64
12  Platform_Support_For_Learning                                           24 non-null    int64
13  Skills_Acquired_Relevant_To_Career_Aspiration                         24 non-null    int64
14  Course_Engaging                                                         24 non-null    int64
15  Motivation_To_Learn_More                                                24 non-null    int64
16  Challenges_Faced                                                         24 non-null    object
17  Confidence_In_Applying_Skills                                           24 non-null    int64
18  Platform_Contribution_To_Learning                                       24 non-null    object
19  Course_Recommendation_To_Others                                         24 non-null    object
20  Subject_Knowledge_After                                                  24 non-null    object
21  Understanding_Of_Topic_After                                             24 non-null    int64
22  Assessment_Completed                                                    24 non-null    int64
23  First_Attempt_Score                                                     24 non-null    object
24  Number_Of_Attempts                                                      24 non-null    int64
25  Hours_Taken_To_Complete_Course                                          24 non-null    float64
26  First_Attempt_Score_No                                                  24 non-null    int64
27  Understanding_Of_Topic_After_Label                                      24 non-null    object
28  Platform_Experience_Label                                                24 non-null    object
29  Ease_of_Navigation_Label                                                 24 non-null    object
dtypes: float64(2), int64(13), object(15)
memory usage: 5.8+ KB

```

```

In [26]: platform_data = df[['Ease_of_Navigation', 'Course_Content_Organisation', 'Usability_Of_Interactive_Elements', 'Platform_Support_For_Learning']]
platform_data.describe()

```

```

Out[26]:

```

	Ease_of_Navigation	Course_Content_Organisation	Usability_Of_Interactive_Elements	Platform_Support_For_Learning
count	24.000000	24.000000	24.000000	24.000000
mean	4.083333	4.333333	3.916667	4.000000
std	0.928611	0.701964	1.017955	0.816497
min	2.000000	3.000000	1.000000	3.000000
25%	3.000000	4.000000	3.750000	3.500000
50%	4.000000	4.000000	4.000000	4.000000
75%	5.000000	5.000000	5.000000	5.000000
max	5.000000	5.000000	5.000000	5.000000

```

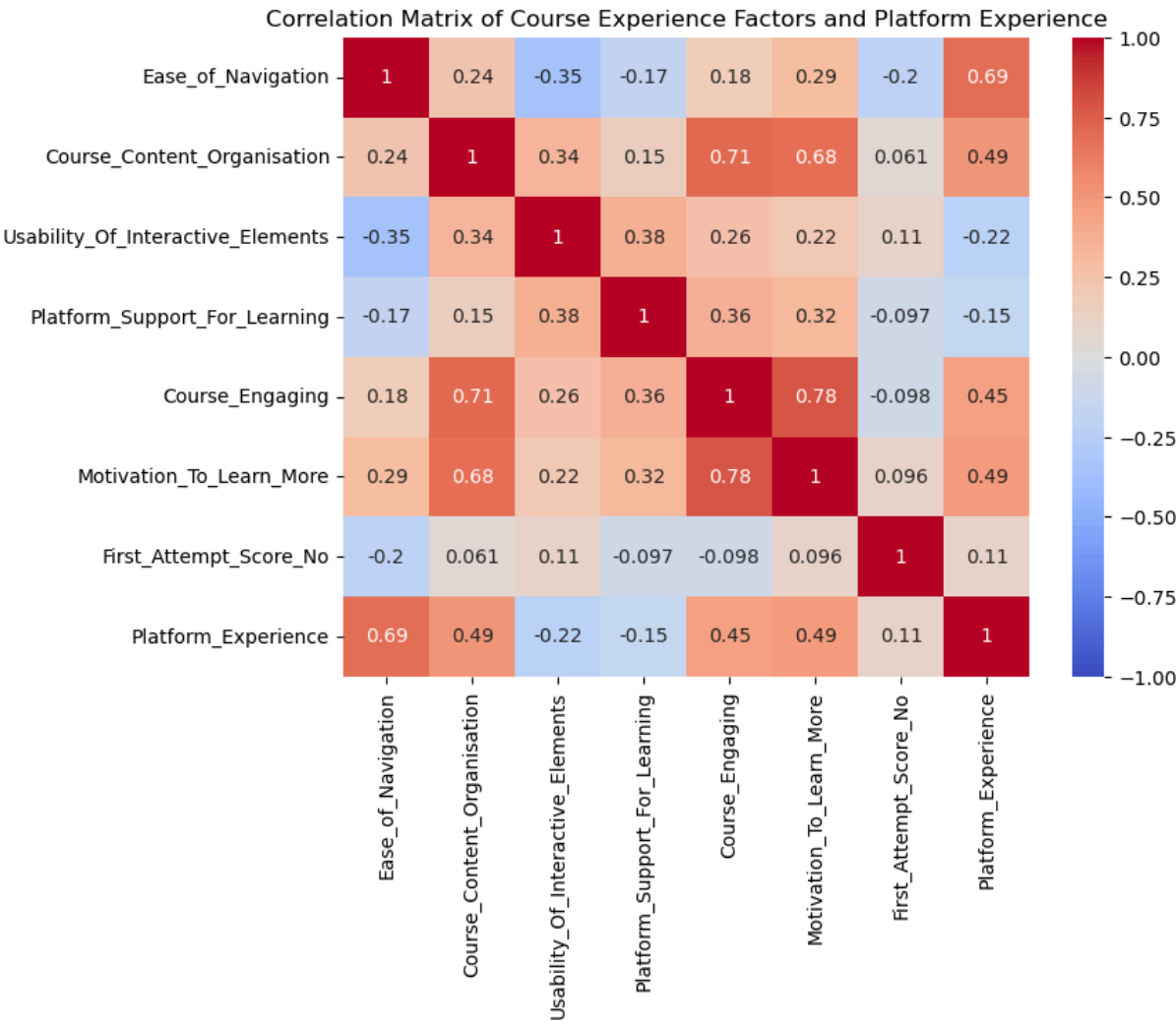
In [27]: # Compute the Spearman correlation matrix
correlation_matrix = df[['Ease_of_Navigation', 'Course_Content_Organisation', 'Usability_Of_Interactive_Elements', 'Platform_Support_For_Learning']].corr()

# Display the correlation matrix as a heatmap
plt.figure(figsize=(8, 6))

```



```
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', vmin=-1, vmax=1)
plt.title('Correlation Matrix of Course Experience Factors and Platform Experience')
plt.show()
```



```
In [28]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 24 entries, 0 to 23
Data columns (total 30 columns):
#   Column                                     Non-Null Count  Dtype
---  -
0   DateTime                                 24 non-null     object
1   Email                                   24 non-null     object
2   Student_Name                           24 non-null     object
3   StudentID                             24 non-null     object
4   Date                                   24 non-null     object
5   Course_Title                           24 non-null     object
6   Course_Duration_Hours                  24 non-null     float64
7   Subject_Knowledge_Before               24 non-null     object
8   Platform_Experience                    24 non-null     int64
9   Ease_of_Navigation                     24 non-null     int64
10  Course_Content_Organisation             24 non-null     int64
11  Usability_Of_Interactive_Elements       24 non-null     int64
12  Platform_Support_For_Learning           24 non-null     int64
13  Skills_Acquired_Relevant_To_Career_Aspiration 24 non-null     int64
14  Course_Engaging                        24 non-null     int64
15  Motivation_To_Learn_More                24 non-null     int64
16  Challenges_Faced                        24 non-null     object
17  Confidence_In_Applying_Skills           24 non-null     int64
18  Platform_Contribution_To_Learning       24 non-null     object
19  Course_Recommendation_To_Others        24 non-null     object
20  Subject_Knowledge_After                 24 non-null     object
21  Understanding_Of_Topic_After            24 non-null     int64
22  Assessment_Completed                    24 non-null     int64
23  First_Attempt_Score                     24 non-null     object
24  Number_Of_Attempts                     24 non-null     int64
25  Hours_Taken_To_Complete_Course          24 non-null     float64
26  First_Attempt_Score_No                  24 non-null     int64
27  Understanding_Of_Topic_After_Label       24 non-null     object
28  Platform_Experience_Label               24 non-null     object
29  Ease_of_Navigation_Label                24 non-null     object
dtypes: float64(2), int64(13), object(15)
memory usage: 5.8+ KB
```

```
In [29]: # Mapping of integers to descriptive labels
relevance_mapping = {
    1: 'Not Relevant',
    2: 'Slightly Relevant',
    3: 'Moderately Relevant',
    4: 'Very Relevant',
    5: 'Extremely Relevant'
}

# Apply the mapping to the column
df['Skills_Acquired_Relevant_Label'] = df['Skills_Acquired_Relevant_To_Career_Aspiration'].map(

# Set the plot size
plt.figure(figsize=(10, 4))

# Plot the bar chart with horizontal bars and adjust the bar width
ax = sns.countplot(data=df, y='Skills_Acquired_Relevant_Label', order=relevance_mapping.values(

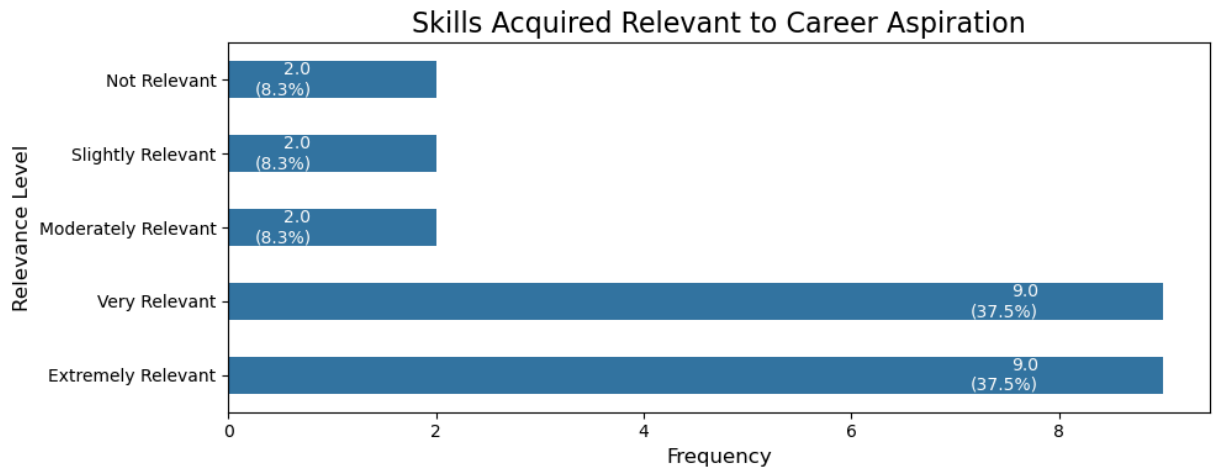
# Calculate the total number of responses
total = len(df)

# Add counts and percentages inside the bars
for p in ax.patches:
    count = p.get_width()
    percentage = '{:.1f}%'.format(100 * count / total)
    ax.annotate(f'{count}\n({percentage})',
                (count - 0.05 * total, p.get_y() + p.get_height() / 2.),
                ha='right', va='center',
                fontsize=10, color='white')

# Add title and labels
plt.title('Skills Acquired Relevant to Career Aspiration', fontsize=16)
plt.ylabel('Relevance Level', fontsize=12)
plt.xlabel('Frequency', fontsize=12)

# Show the plot
```

```
plt.tight_layout()
plt.show()
```



```
In [30]: # Group the data by the course
grouped = df.groupby('Course_Title')

# Iterate over each course group
for course_name, group in grouped:

    # Set the plot size for each individual course
    plt.figure(figsize=(10, 4))

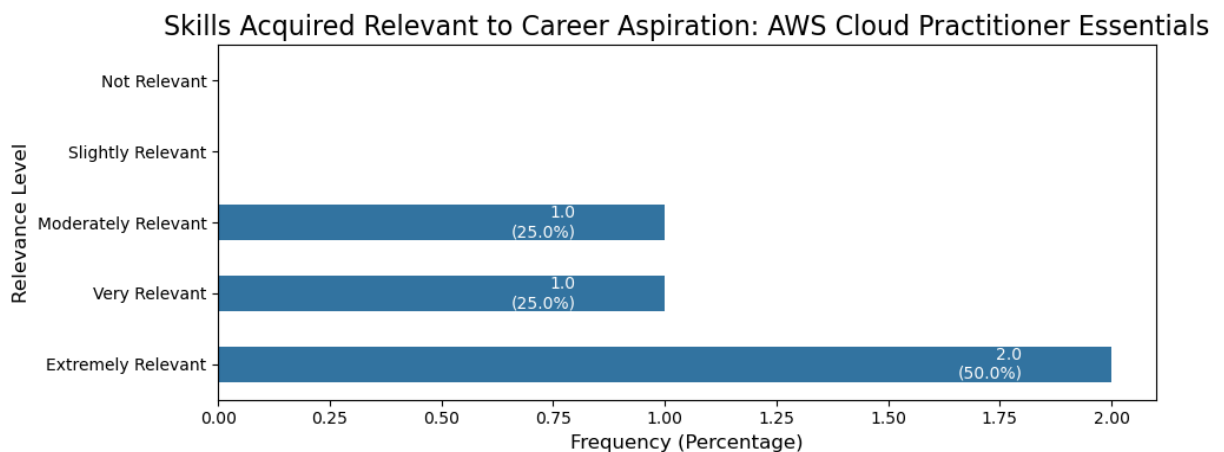
    # Plot the bar chart with horizontal bars for the current course group
    ax = sns.countplot(data=group, y='Skills_Acquired_Relevant_Label',
                       order=relevance_mapping.values(), width=0.5)

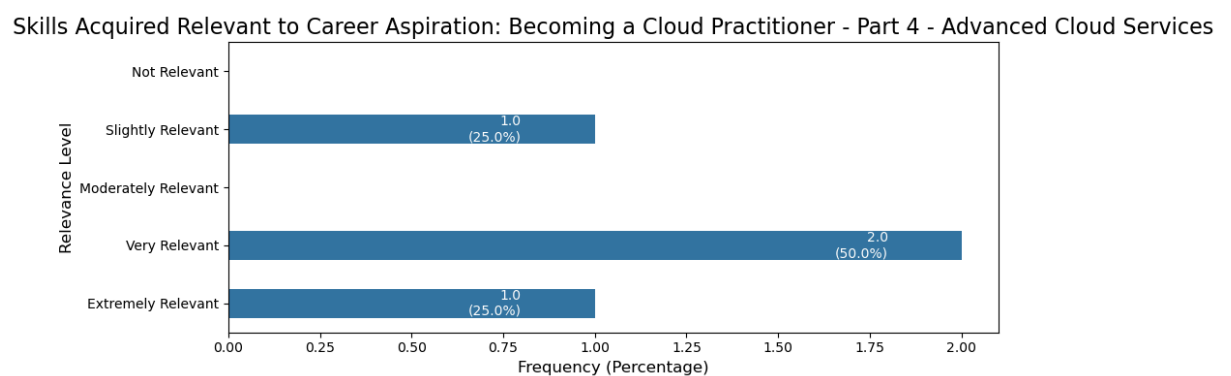
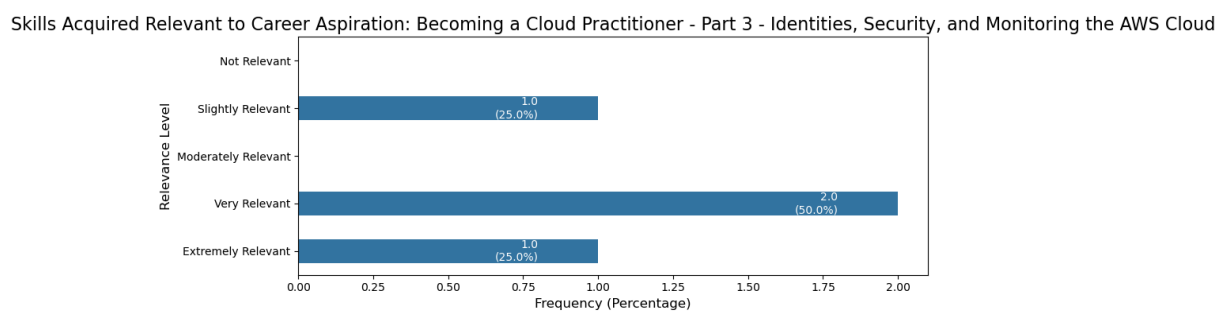
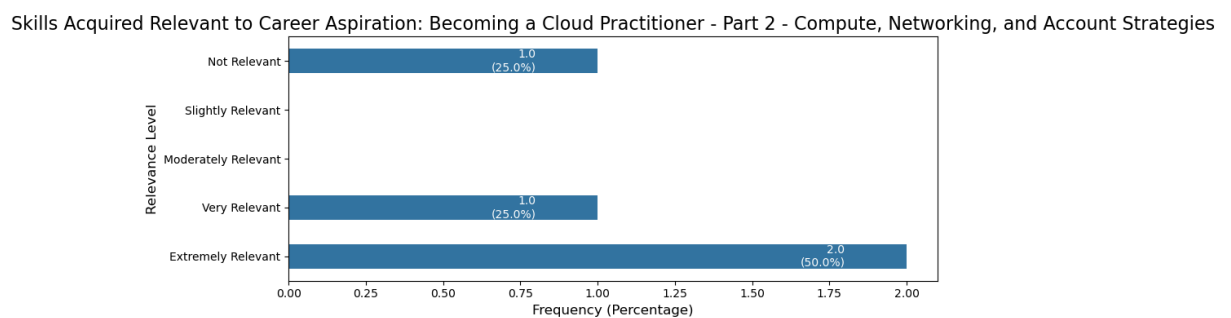
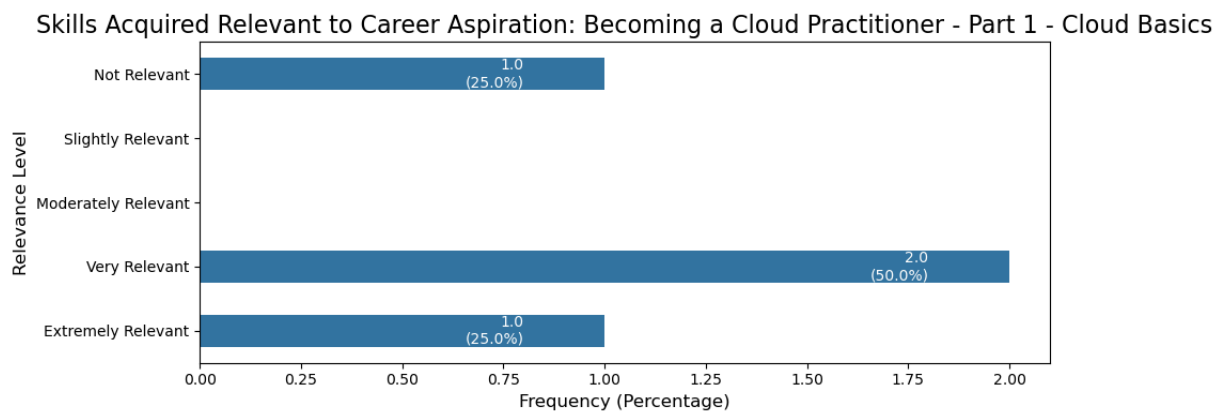
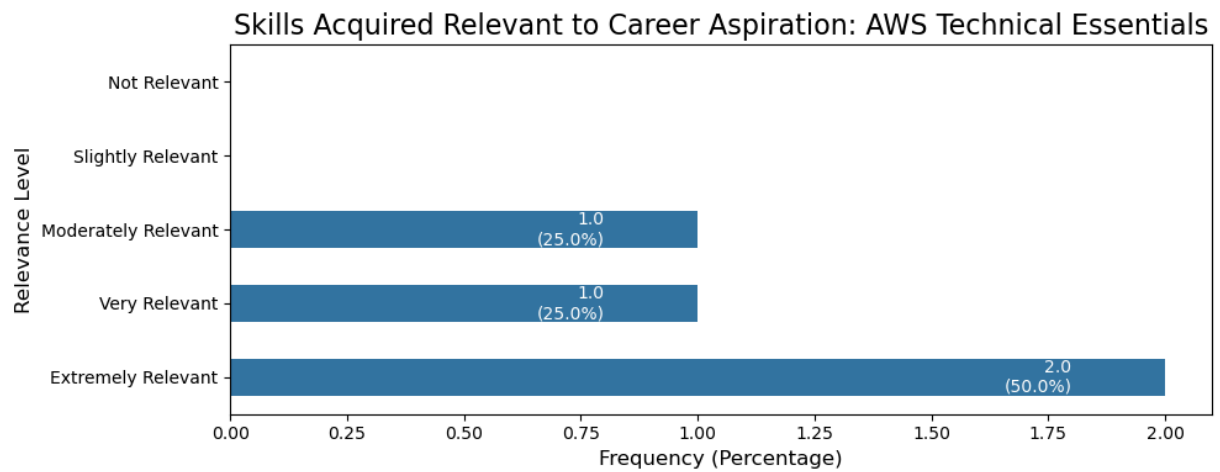
    # Calculate the total number of responses for the current course
    total = len(group)

    # Add counts and percentages inside the bars
    for p in ax.patches:
        count = p.get_width()
        percentage = '{:.1f}%'.format(100 * count / total)
        ax.annotate(f'{count}\n({percentage})',
                    (count - 0.05 * total, p.get_y() + p.get_height() / 2.),
                    ha='right', va='center',
                    fontsize=10, color='white')

    # Add title and labels
    plt.title(f'Skills Acquired Relevant to Career Aspiration: {course_name}', fontsize=16)
    plt.ylabel('Relevance Level', fontsize=12)
    plt.xlabel('Frequency (Percentage)', fontsize=12)

    # Adjust layout and show the plot for the current course
    plt.tight_layout()
    plt.show()
```





```
In [31]: # Mapping of integers to descriptive labels
confidence_mapping = {
    1: 'Not Confident',
```

```

2: 'Slightly Confident',
3: 'Moderately Confident',
4: 'Confident',
5: 'Higly Confident'
}

# Apply the mapping to the column
df['Confidence_In_Applying_Skills_Label'] = df['Confidence_In_Applying_Skills'].map(confidence_

```

```

In [32]: # Set the plot size
plt.figure(figsize=(10, 4))

# Plot the bar chart with horizontal bars and adjust the bar width
ax = sns.countplot(data=df, y='Confidence_In_Applying_Skills_Label', order=confidence_mapping.v

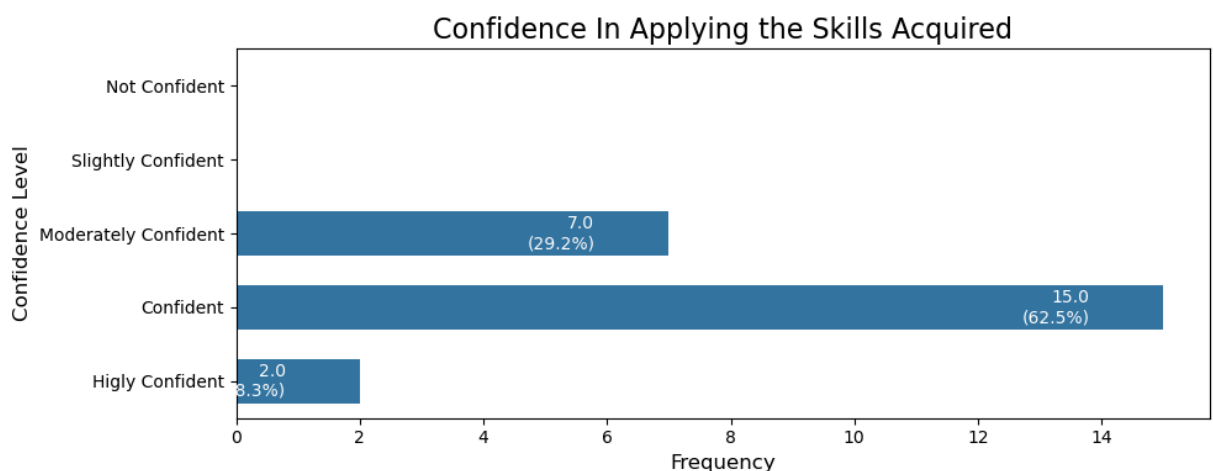
# Calculate the total number of responses
total = len(df)

# Add counts and percentages inside the bars
for p in ax.patches:
    count = p.get_width()
    percentage = '{:.1f}%'.format(100 * count / total)
    ax.annotate(f'{count}\n({percentage})',
                (count - 0.05 * total, p.get_y() + p.get_height() / 2.),
                ha='right', va='center',
                fontsize=10, color='white')

# Add title and labels
plt.title('Confidence In Applying the Skills Acquired', fontsize=16)
plt.ylabel('Confidence Level', fontsize=12)
plt.xlabel('Frequency', fontsize=12)

# Show the plot
plt.tight_layout()
plt.show()

```



```

In [33]: # Descriptive statistics for First Attempt Score
print("First Score Attempt Analysis")
print(df['First_Attempt_Score_No'].describe())

```

```

First Score Attempt Analysis
count    24.00000
mean     74.62500
std      19.27786
min      33.00000
25%      64.25000
50%      78.00000
75%      90.50000
max      100.00000
Name: First_Attempt_Score_No, dtype: float64

```

```

In [34]: # Descriptive statistics for First Attempt Score
print("Number of Attempt Analysis")
print(df['Number_Of_Attempts'].describe())

```

```

Number of Attempt Analysis
count    24.000000
mean      1.500000
std       0.722315
min       1.000000
25%      1.000000
50%      1.000000
75%      2.000000
max       4.000000
Name: Number_Of_Attempts, dtype: float64

```

```

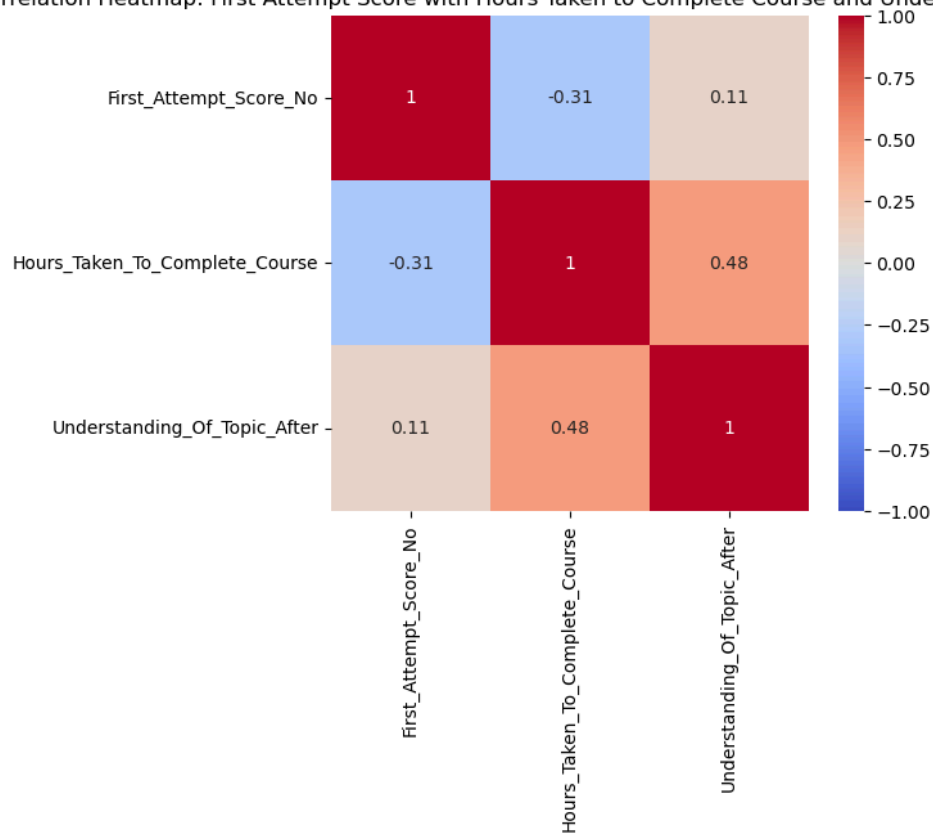
In [35]: # Create a DataFrame with the three columns
data = df[['First_Attempt_Score_No', 'Hours_Taken_To_Complete_Course', 'Understanding_Of_Topic_

# Compute the correlation matrix
correlation_matrix = data.corr()

# Create the heatmap
plt.figure(figsize=(6, 5))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', vmin=-1, vmax=1)
plt.title('Correlation Heatmap: First Attempt Score with Hours Taken to Complete Course and Und
plt.show()

```

Correlation Heatmap: First Attempt Score with Hours Taken to Complete Course and Understanding of the Topic



```

In [36]: df.info()

```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 24 entries, 0 to 23
Data columns (total 32 columns):
#      Column                                     Non-Null Count  Dtype
---  -
0     DateTime                                     24 non-null     object
1     Email                                       24 non-null     object
2     Student_Name                             24 non-null     object
3     StudentID                                24 non-null     object
4     Date                                       24 non-null     object
5     Course_Title                             24 non-null     object
6     Course_Duration_Hours                   24 non-null     float64
7     Subject_Knowledge_Before                24 non-null     object
8     Platform_Experience                     24 non-null     int64
9     Ease_of_Navigation                      24 non-null     int64
10    Course_Content_Organisation             24 non-null     int64
11    Usability_Of_Interactive_Elements       24 non-null     int64
12    Platform_Support_For_Learning           24 non-null     int64
13    Skills_Acquired_Relevant_To_Career_Aspiration 24 non-null     int64
14    Course_Engaging                         24 non-null     int64
15    Motivation_To_Learn_More                24 non-null     int64
16    Challenges_Faced                        24 non-null     object
17    Confidence_In_Applying_Skills           24 non-null     int64
18    Platform_Contribution_To_Learning       24 non-null     object
19    Course_Recommendation_To_Others         24 non-null     object
20    Subject_Knowledge_After                 24 non-null     object
21    Understanding_Of_Topic_After            24 non-null     int64
22    Assessment_Completed                   24 non-null     int64
23    First_Attempt_Score                     24 non-null     object
24    Number_Of_Attempts                      24 non-null     int64
25    Hours_Taken_To_Complete_Course          24 non-null     float64
26    First_Attempt_Score_No                  24 non-null     int64
27    Understanding_Of_Topic_After_Label       24 non-null     object
28    Platform_Experience_Label               24 non-null     object
29    Ease_of_Navigation_Label                24 non-null     object
30    Skills_Acquired_Relevant_Label          24 non-null     object
31    Confidence_In_Applying_Skills_Label     24 non-null     object
dtypes: float64(2), int64(13), object(17)
memory usage: 6.1+ KB
```

```
In [37]: # Ensure pandas shows the full text in the columns
pd.set_option('display.max_colwidth', None)

# Print the column (Challenges_Faced)
print(df['Challenges_Faced'])
```

0

In the module there were so many information to follow and needed to memorise to do the exercise part. But when I figured out there is a Hint button to get hints that heled me to overcome the c hallenges.

1 I started with simulations since it was placed at the beginning of the course and had some difficulty in understanding the instructions. But as I progress through the course I realized th at we can go to the simulation after learning the lessons. After learning the lessons and going back to simulation helped in working with the simulation. Also since I like learning by video ba sed courses, I had some challenges in going through the course. I started taking small notes as I understand the lessons and that helped me.

2

at first, I was little confused with the content organisation due to which I completed the simul ation part before the actual course had begun. Later, I realised it and then connected my simula tion experience with the actual course content and I could connect the dots well.

3

I did not face any significant challenges as it was my second time accessing a course to learn.

4

Since I have taken part 1, this time I had not faced much challenge. The only challenge is takin g the course at one go. It is better to complete the course at one go and do the assessment sinc e the learning stays in my memory.

5

Too much information to think and got confused much time while doing the exercise

6

Initially, navigation was quite frustrating as I wasn't able to go back, and I was closing the t abs to get out but later I figured it out .

7

No specific challenges are faced knowing the organisation of the content and having some backgro und in on premise infrastructure management.

8

While doing the quiz I found most of the questions are same and the choices are all relevant. I had to go back to modules to double check the answer.

9

No specific challenges are faced. As I progress through the course, I realized that it is better to take the course on one go as all the concepts are related.

10

Initially, I found the Data Analytics in the AWS Cloud quite difficult but when I went through e verything and put my mind on it I was confident enough while doing the exercise.

11

Initially, navigation was quite frustrating as I wasn't able to go back and I was closing the ta bs to get out but later I figured it out.

12

UI is not good and I don't feel comfortable with it.

13

I have got familiar with the UI now so it's easy now but initially, it was not easy

14

sometimes, its very difficult to understand certain concepts from just reading, for which I have to search in google to get clearer understanding.

15

I didn't encounter any challenge for this course

16

The only challenge I face was the duration of the course. Since it was a 7 hours course it took me some time to complete the course. Learning few modules divided over different days and taking notes helped me complete the course successfully.

17

I did not face any significant challenges during the learning process.

18

I had no knowledge before about AWS root user, AWS root user credentials, user guide, IAM and IA M identities. It took me a long time to understand the course.

19

This course was easy, with fewer quiz questions and a little challenging.

20

Courses are getting lengthier and have to cut the sleep time

21

Courses are getting lengthier and have to cut the sleep time.

22

The only challenge I face was the time needed to complete the course. The course duration is 4 h ours but it took me more than 10 hours. Since the course is divided into different modules, I tr ied completing 1 or 2 modules everytime I continued with the course.

23

The only challenge was the duration of the course as it was very a very long course of 7 hours. I tried to learn the courses little at a time to complete the course.

Name: Challenges_Faced, dtype: object

```
In [38]: # Ensure pandas shows the full text in the columns
pd.set_option('display.max_colwidth', None)
```



```
# Print the column (Course_Recommendation_To_Others)
print(df['Course_Recommendation_To_Others'])
```

```
0          Yes, Because the platform will help all the beginners and no had
no knowledge about AWS platform can learn a lot and use the learning experience in their future
career and job career.
1
2
3
4          I would only recommend th
e course to who has basic knowledge of how computer networks works, OS, servers and are interest
ed in Cloud Computing.
5
6          I think the AWS platform is really effective for learning about AWS and the cloud.
7
8          No, it was of no use as it is only applicable to those who want to work in AWS.
9
10         Yes, this course will be very important to someone who is interested in Cloud Computing.
11         yes I will definitely recommend this platform to all my friends who are interested to do I
T course in the future so that they will have a basic idea about the AWS and cloud etc before th
ey start their course.
12
13         Yes if someone is interested in Cloud Computing.
14         Yes, I will r
ecommend this course to my friends because there is a lot to learn and we can use the skill in o
ur future job careers.
15
16         No, it was of no use as it is only applicable to those who want to work in AWS.
17
18         It depends on the people asking me these questions if they are into Amazon and AWS then yes.
19
20         If they are into AWS, yes.
21         yes because t
his course teaches lots of new things related to cloud computing and would be a must have skill
for all IT individual.
22
23         Yes as the course contents are relatable if you are working for a company who uses AWS services.
24
25         Yes if someone is interested to become Cloud Engineer.
26
27         yes because this course give us the basics of AWS and the contents are also easier to follow
28
29         yes
30         yes
31         It is recommended to people who are going to work in Amazon or people who are going to network s
ide.
32
33         It is recommended to people who are going to work in Amazon or people who are going to network s
ide.
34
35         Yes, to anyone who are trying to get into Cloud Computing profession (AWS).
36
37         yes
38         Name: Course_Recommendation_To_Others, dtype: object
```

In []: