

In [1]:

```
import pandas as pd
```

In [4]:

```
df = pd.read_csv('survey_ai.csv')
```

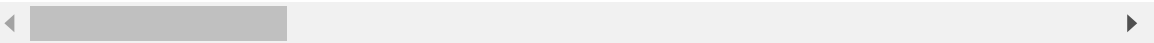
In [5]:

```
df.head()
```

Out[5]:

	ID	Q1.AI_knowledge	Q2.AI_sources	Q2#1.Internet	Q2#2.Books/Papers	Q2#3.Soci
0	1	8	Internet;Books/Scientific papers (physical/onl...	1	1	
1	2	7	Internet;Social media	1	0	
2	3	5	Internet;Books/Scientific papers (physical/onl...	1	1	
3	4	5	Internet;Social media	1	0	
4	5	4	Internet	1	0	

5 rows × 35 columns



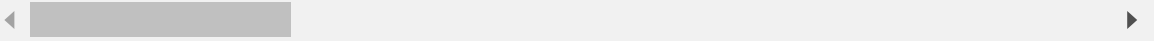
In [6]:

```
df.tail()
```

Out[6]:

	ID	Q1.AI_knowledge	Q2.AI_sources	Q2#1.Internet	Q2#2.Books/Papers	Q2#3.Social_med
86	87	8	Internet	1	0	
87	88	6	Social media	0	0	
88	89	9	Books/Scientific papers (physical/online format)	0	1	
89	90	2	I don't inform myself about AI	0	0	
90	91	2	Discussions with family/friends	0	0	

5 rows × 35 columns



In [7]:

```
df.shape
```

Out[7]:

```
(91, 35)
```

In [8]:

```
df.columns
```

Out[8]:

```
Index(['ID', 'Q1.AI_knowledge', 'Q2.AI_sources', 'Q2#1.Internet',  
      'Q2#2.Books/Papers', 'Q2#3.Social_media', 'Q2#4.Discussions',  
      'Q2#5.NotInformed', 'Q3#1.AI_dehumanization', 'Q3#2.Job_replacemen  
t',  
      'Q3#3.Problem_solving', 'Q3#4.AI_rulling_society', 'Q4#1.AI_costl  
y',  
      'Q4#2.Economic_crisis', 'Q4#3.Economic_growth', 'Q4#4.Job_loss',  
      'Q5Feelings', 'Q6.Domains', 'Q6#1.Education', 'Q6#2.Medicine',  
      'Q6#3.Agriculture', 'Q6#4.Constructions', 'Q6#5.Marketing',  
      'Q6#6.Administration', 'Q6#7.Art', 'Q7.Utility_grade',  
      'Q8.Advantage_teaching', 'Q9.Advantage_learning',  
      'Q10.Advantage_evaluation ', 'Q11.Disadvantage_educational_proces  
s',  
      'Q12.Gender', 'Q13.Year_of_study', 'Q14.Major', 'Q15.Passed_exams',  
      'Q16.GPA'],  
      dtype='object')
```

In [9]:

df.info()

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 91 entries, 0 to 90
Data columns (total 35 columns):
#   Column                                     Non-Null Count  Dtype
---  -
0   ID                                           91 non-null     int64
1   Q1.AI_knowledge                             91 non-null     int64
2   Q2.AI_sources                              91 non-null     object
3   Q2#1.Internet                             91 non-null     int64
4   Q2#2.Books/Papers                         91 non-null     int64
5   Q2#3.Social_media                         91 non-null     int64
6   Q2#4.Discussions                          91 non-null     int64
7   Q2#5.NotInformed                          91 non-null     int64
8   Q3#1.AI_dehumanization                    91 non-null     int64
9   Q3#2.Job_replacement                      91 non-null     int64
10  Q3#3.Problem_solving                      91 non-null     int64
11  Q3#4.AI_rulling_society                   91 non-null     int64
12  Q4#1.AI_costly                            91 non-null     int64
13  Q4#2.Economic_crisis                     91 non-null     int64
14  Q4#3.Economic_growth                     91 non-null     int64
15  Q4#4.Job_loss                             91 non-null     int64
16  Q5.Feelings                              91 non-null     int64
17  Q6.Domains                               91 non-null     object
18  Q6#1.Education                           91 non-null     int64
19  Q6#2.Medicine                            91 non-null     int64
20  Q6#3.Agriculture                         91 non-null     int64
21  Q6#4.Constructions                       91 non-null     int64
22  Q6#5.Marketing                           91 non-null     int64
23  Q6#6.Administration                      91 non-null     int64
24  Q6#7.Art                                 91 non-null     int64
25  Q7.Utility_grade                         91 non-null     int64
26  Q8.Advantage_teaching                    91 non-null     int64
27  Q9.Advantage_learning                    91 non-null     int64
28  Q10.Advantage_evaluation                  91 non-null     int64
29  Q11.Disadvantage_educational_process     91 non-null     int64
30  Q12.Gender                              91 non-null     int64
31  Q13.Year_of_study                        91 non-null     int64
32  Q14.Major                                91 non-null     int64
33  Q15.Passed_exams                         91 non-null     int64
34  Q16.GPA                                  91 non-null     float64
dtypes: float64(1), int64(32), object(2)
memory usage: 25.0+ KB

```

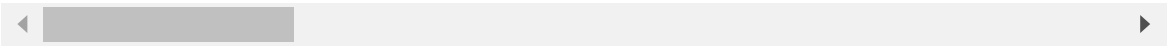
In [10]:

```
df.describe()
```

Out[10]:

	ID	Q1.AI_knowledge	Q2#1.Internet	Q2#2.Books/Papers	Q2#3.Social_media	Q2#4.Social_media
count	91.00000	91.000000	91.000000	91.000000	91.000000	91.000000
mean	46.00000	5.912088	0.813187	0.351648	0.439560	0.439560
std	26.41338	1.970044	0.391921	0.480130	0.499083	0.499083
min	1.00000	1.000000	0.000000	0.000000	0.000000	0.000000
25%	23.50000	5.000000	1.000000	0.000000	0.000000	0.000000
50%	46.00000	6.000000	1.000000	0.000000	0.000000	0.000000
75%	68.50000	7.000000	1.000000	1.000000	1.000000	1.000000
max	91.00000	10.000000	1.000000	1.000000	1.000000	1.000000

8 rows × 33 columns



In [11]:

```
df.nunique()
```

Out[11]:

ID	91
Q1.AI_knowledge	10
Q2.AI_sources	13
Q2#1.Internet	2
Q2#2.Books/Papers	2
Q2#3.Social_media	2
Q2#4.Discussions	2
Q2#5.NotInformed	2
Q3#1.AI_dehumanization	5
Q3#2.Job_replacement	5
Q3#3.Problem_solving	5
Q3#4.AI_rulling_society	5
Q4#1.AI_costly	5
Q4#2.Economic_crisis	5
Q4#3.Economic_growth	4
Q4#4.Job_loss	5
Q5.Feelings	4
Q6.Domains	44
Q6#1.Education	2
Q6#2.Medicine	2
Q6#3.Agriculture	2
Q6#4.Constructions	2
Q6#5.Marketing	2
Q6#6.Administration	2
Q6#7.Art	2
Q7.Utility_grade	9
Q8.Advantage_teaching	3
Q9.Advantage_learning	3
Q10.Advantage_evaluation	3
Q11.Disadvantage_educational_process	4
Q12.Gender	2
Q13.Year_of_study	2
Q14.Major	3
Q15.Passed_exams	2
Q16.GPA	10
dtype:	int64

In [16]:

```
for i in df.columns:
    print(i)
    print(df[i].unique())
    print('\n')
```

```
ID
[ 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 2
 4
 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 4
 8
 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 7
 2
 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91]
```

Q1.AI_knowledge

```
[ 8  7  5  4  6  9  1  3 10  2]
```

Q2.AI_sources

```
['Internet;Books/Scientific papers (physical/online format)'
 'Internet;Social media' 'Internet'
 'Internet;Discussions with family/friends'
 'Books/Scientific papers (physical/online format);Social media;Discuss
 .
```

In [17]:

```
for i in df.columns:
    print(i)
    print(df[i].value_counts())
    print('\n')
```

```
ID
1      1
69     1
67     1
66     1
65     1
..
29     1
28     1
27     1
26     1
91     1
Name: ID, Length: 91, dtype: int64
```

Q1.AI_knowledge

```
7      17
5      17
6      17
..
```

```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
from IPython import get_ipython
import warnings
warnings.filterwarnings("ignore")
```

```
for i in df.columns:
    plt.figure(figsize=(15,6))
    sns.countplot(df[i], data = df, palette = 'hls')
    plt.show()
```

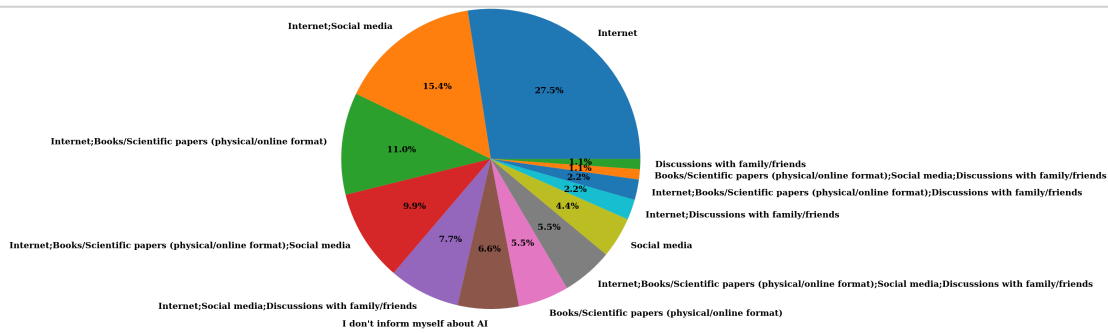
In [38]:

```

for i in df.columns:
    plt.figure(figsize=(30,20))
    plt.pie(df[i].value_counts(), labels=df[i].value_counts().index,
            autopct='%1.1f%%', textprops={ 'fontsize': 25,
                                            'color': 'black',
                                            'weight': 'bold',
                                            'family': 'serif' })

hfont = {'fontname':'serif', 'weight': 'bold'}
plt.title(i, size=20, **hfont)
plt.show()

```



1



In [22]:

```

for i in df.columns:
    plt.figure(figsize=(15,6))
    sns.histplot(df[i], bins = 10, kde = True, palette = 'hls')
    plt.xticks(rotation = 90)
    plt.show()

```



In [25]:

```

num_cols = df.select_dtypes(include=[np.number])

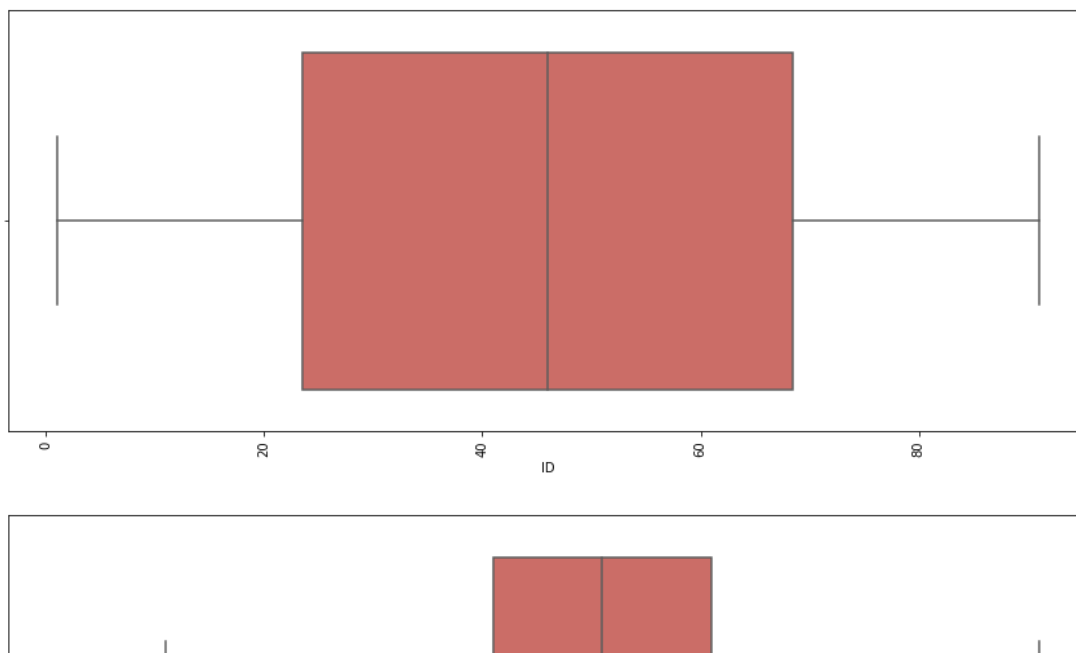
```

In [26]:

```

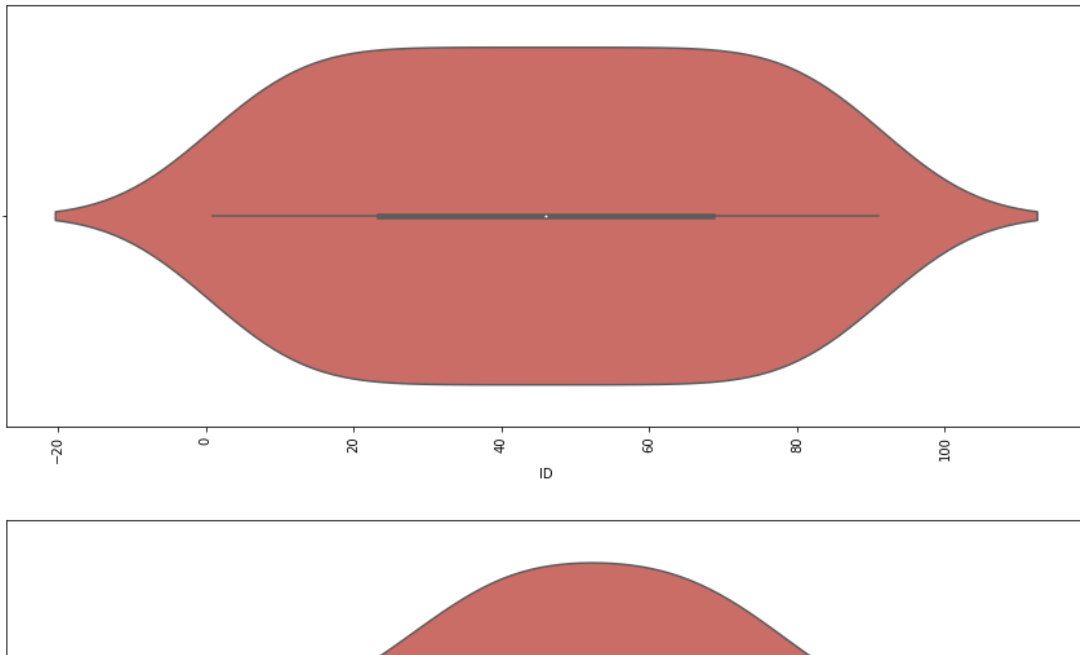
for i in num_cols.columns:
    plt.figure(figsize=(15,6))
    sns.boxplot(num_cols[i], data = df, palette = 'hls')
    plt.xticks(rotation = 90)
    plt.show()

```



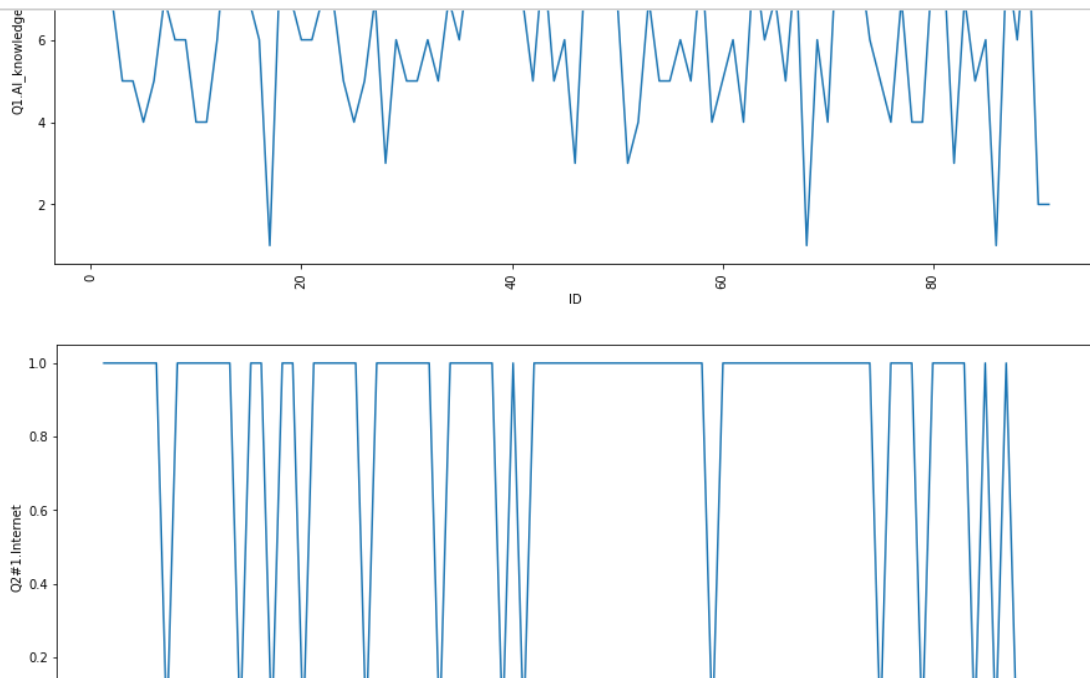
In [27]:

```
for i in num_cols.columns:  
    plt.figure(figsize=(15,6))  
    sns.violinplot(num_cols[i], data = df, palette = 'hls')  
    plt.xticks(rotation = 90)  
    plt.show()
```



In [28]:

```
for i in num_cols.columns:  
    for j in num_cols.columns:  
        plt.figure(figsize=(15,6))  
        sns.lineplot(x = num_cols[i], y = num_cols[j], data = df, palette = 'hls')  
        plt.xticks(rotation = 90)  
        plt.show()
```

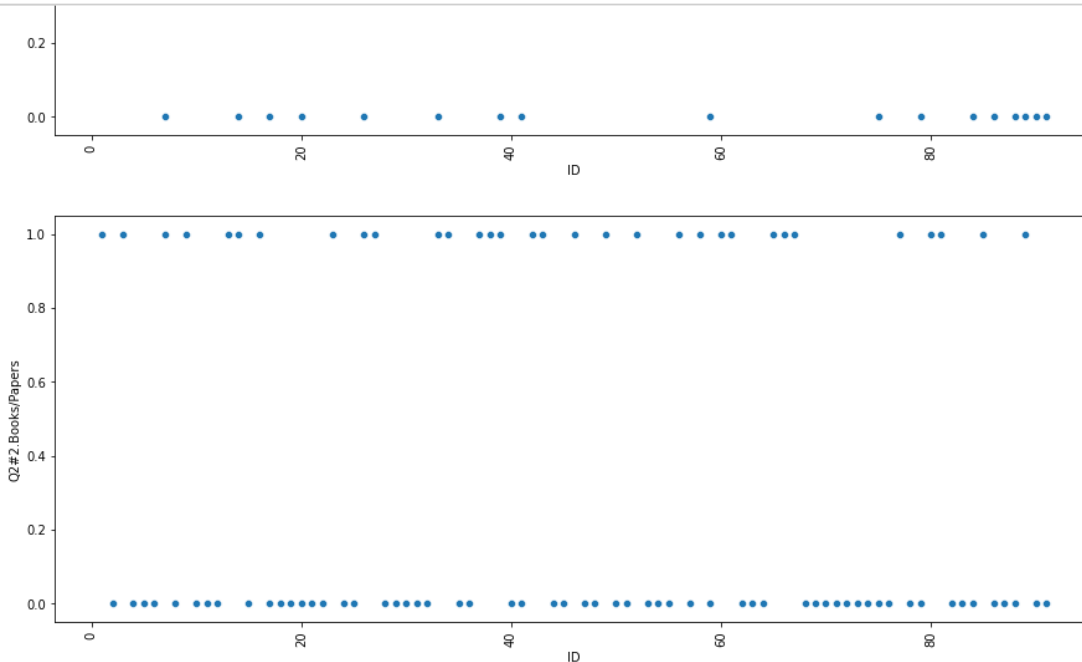


In [29]:

```

for i in num_cols.columns:
    for j in num_cols.columns:
        plt.figure(figsize=(15,6))
        sns.scatterplot(x = num_cols[i], y = num_cols[j], data = df, palette = 'hls')
        plt.xticks(rotation = 90)
        plt.show()

```

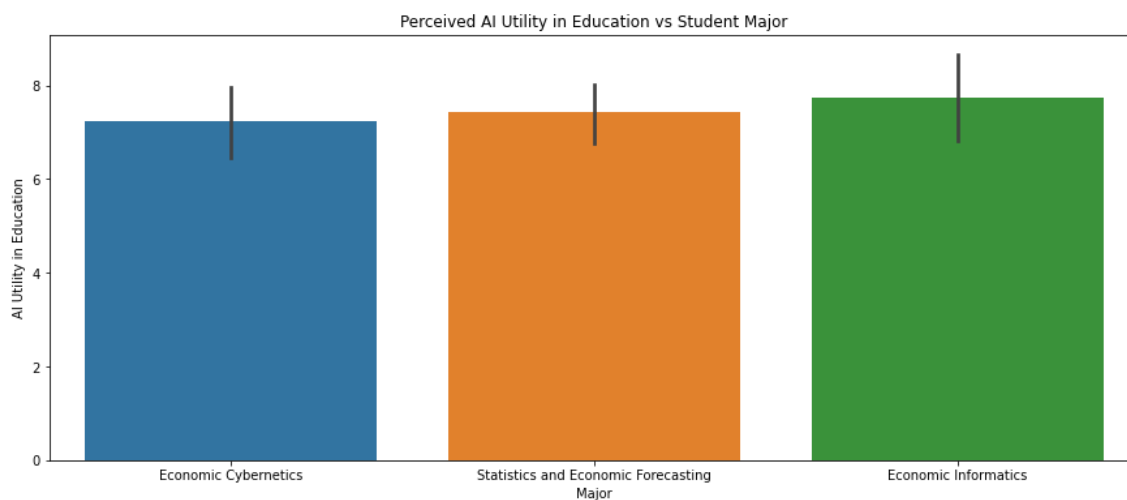


In [34]:

```

plt.figure(figsize=(15,6))
plot = sns.barplot(data=df[['Q14.Major', 'Q7.Utility_grade']],
                  x='Q14.Major', y='Q7.Utility_grade')
plot.set(xlabel='Major', ylabel='AI Utility in Education',
        title='Perceived AI Utility in Education vs Student Major')
plot.set_xticklabels(['Economic Cybernetics', 'Statistics and Economic Forecasting',
                    'Economic Informatics'], rotation=0)
plt.show()

```



In [35]:

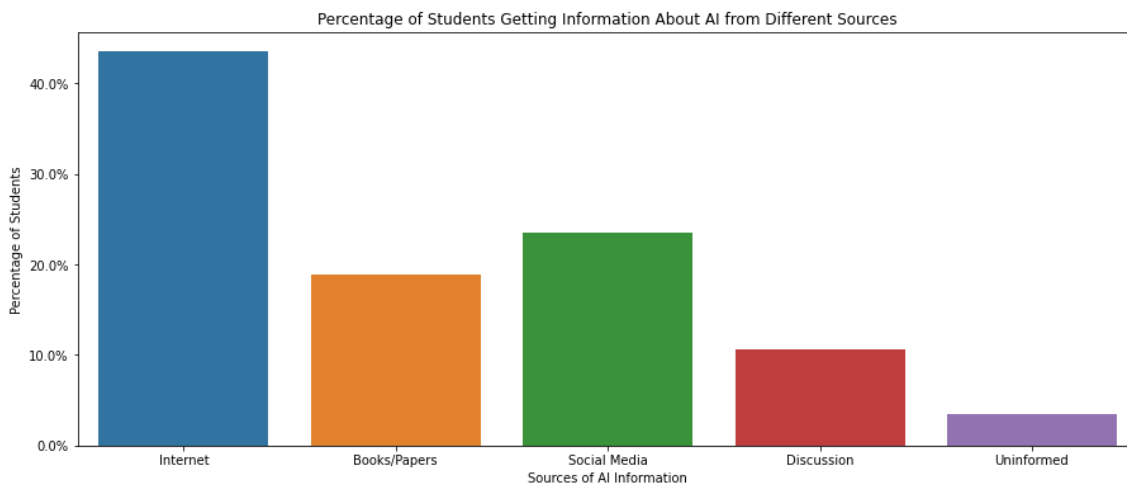
```
source_sum = pd.DataFrame(df[['Q2#1.Internet', 'Q2#2.Books/Papers',
                             'Q2#3.Social_media', 'Q2#4.Discussions',
                             'Q2#5.NotInformed']].sum())

# Getting percentage of sample size
sample_size = source_sum[0].sum()

# Getting percentage column
source_sum[1] = source_sum[0] / sample_size * 100
```

In [37]:

```
import matplotlib.ticker as mtick
plt.figure(figsize=(15,6))
ax = sns.barplot(data=source_sum, x=source_sum.index, y=source_sum.columns[1])
ax.set(xlabel='Sources of AI Information', ylabel='Percentage of Students',
       title='Percentage of Students Getting Information About AI from Different Sources')
ax.set_xticklabels(['Internet', 'Books/Papers', 'Social Media', 'Discussion',
                    'Uninformed'], rotation=0)
ax.yaxis.set_major_formatter(mtick.PercentFormatter())
plt.show()
```



In [39]:

```
informed = df.groupby(['Q2#5.NotInformed']).mean()
informed_utility = pd.DataFrame(informed['Q7.Utility_grade'])
informed_utility['Utility_Percentage'] = informed['Q7.Utility_grade'].apply(lambda x : x)
informed_utility.columns
```

Out[39]:

```
Index(['Q7.Utility_grade', 'Utility_Percentage'], dtype='object')
```

In [41]:

```
plt.figure(figsize=(15,6))
sentiment = sns.barplot(data=informed_utility, x=informed_utility.index, y='Q7.Utility_g
sentiment.set(xlabel='Is Student Informed', ylabel='Utility Grade of AI in Education',
              title='Sentiment of AI in Education vs Is Informed on AI')
sentiment.set_xticklabels(['Is Informed', 'Is Not Informed'], rotation=0)
plt.show()
```

