In [1]:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

In [2]:

```
# load the data into pandas dataframes
demographic_data  = pd.read_excel("Table 1.xlsx")
genitive_plurals   = pd.read_excel("Table 2.xlsx")

demographic_data.head()
```

Out[2]:

	Speaker	Religion	Gender	Age	Length of interview (1 =less 30 min, 2 = 30 min to 1 hour,3=1 hour to 2 hours, 4=over two hours)	Village
0	Cases\\Speakers\\B10	Skotadi	Male	53	2	Bonriki
1	Cases\\Speakers\\B11	Drepadian	Female	51	1	Bonriki
2	Cases\\Speakers\\B12	Drepadian	Male	60	2	Bonriki
3	Cases\\Speakers\\B13	Thalassic	Male	45	2	Bonriki
4	Cases\\Speakers\\B14	Thalassic	Male	48	2	Bonriki

In [3]:

```
# extract the 'nid' column from 'speaker' column of table_1
demographic_data['Nid'] = demographic_data['Speaker'].str.split('\\').str[-1]

# drop the 'Speaker' column since it's no longer needed
demographic_data.drop(columns=['Speaker'],inplace=True)
```

In [4]:

```
# Check for missing values in demographic_data
missing_values = demographic_data.isnull().sum()
print(missing_values)

# Check for missing values in genitive_plurals
missing_values = genitive_plurals.isnull().sum()
print(missing_values)
```

```
Religion
0
Gender
0
Age
0
Length of interview ( 1 =less 30 min, 2 = 30 min to 1 hour,3=1 hour to 2 hours, 4=over two hour
s)    0
Village
0
Nid
0
dtype: int64
Nid      0
A7       0
B7       0
C7       0
D7       0
E7       0
F7       0
G7       0
H7       0
I7       0
Total    0
dtype: int64
```

In [5]:

```
# Check the tables
print(demographic_data)
print(genitive_plurals)
```

```
     Religion  Gender  Age  
0     Skotadi    Male   53  \
1   Drepadian  Female   51  
2   Drepadian    Male   60  
3   Thalassic    Male   45  
4   Thalassic    Male   48  
..        ...     ...  ...  
57  Drepadian  Female   67  
58    Skotadi  Female   49  
59  Drepadian  Female   73  
60    Skotadi  Female   54  
61  Drepadian  Female   30  

    Length of interview ( 1 =less 30 min, 2 = 30 min to 1 hour,3=1 hour to 2 hours, 4=over two
hours)  
0                                                   2                                              
\
1                                                   1                                              
2                                                   2                                              
3                                                   2                                              
4                                                   2                                              
..                                                ...                                              
57                                                  3                                              
58                                                  2                                              
59                                                  4                                              
60                                                  2                                              
61                                                  4                                              

       Village  Nid  
0      Bonriki  B10  
1      Bonriki  B11  
2      Bonriki  B12  
3      Bonriki  B13  
4      Bonriki  B14  
..         ...  ...  
57  Nawerewere  Z10  
58  Nawerewere   Z2  
59  Nawerewere   Z4  
60  Nawerewere   Z5  
61  Nawerewere   Z7  

[62 rows x 6 columns]
      Nid  A7  B7  C7  D7  E7  F7  G7  H7  I7  Total
0     B10   2   2   0   0   0   0   0   0   0      4
1     B11   0   0   0   0   0   0   0   0   0      0
2     B12   0   0   0   0   0   0   0   0   0      0
3     B13   0   0   0   0   0   0   0   0   0      0
4     B14   1   1   0   0   0   0   0   0   0      2
..    ...  ..  ..  ..  ..  ..  ..  ..  ..  ..    ...
58     Z2   0   0   0   0   0   0   0   0   0      0
59     Z4   3   1   0   0   1   0   1   0   0      6
60     Z5   0   0   0   0   0   0   0   0   0      0
61     Z7   0   0   0   0   0   0   0   0   0      0
62  Total  98  32   2   1  46   2   9   5   1    196

[63 rows x 11 columns]
```

In [6]:

```
# merge the two tables on the 'Mid' column
merged_table = pd.merge(demographic_data, genitive_plurals, on='Nid')
```

In [7]:

```
merged_table
```

Out[7]:

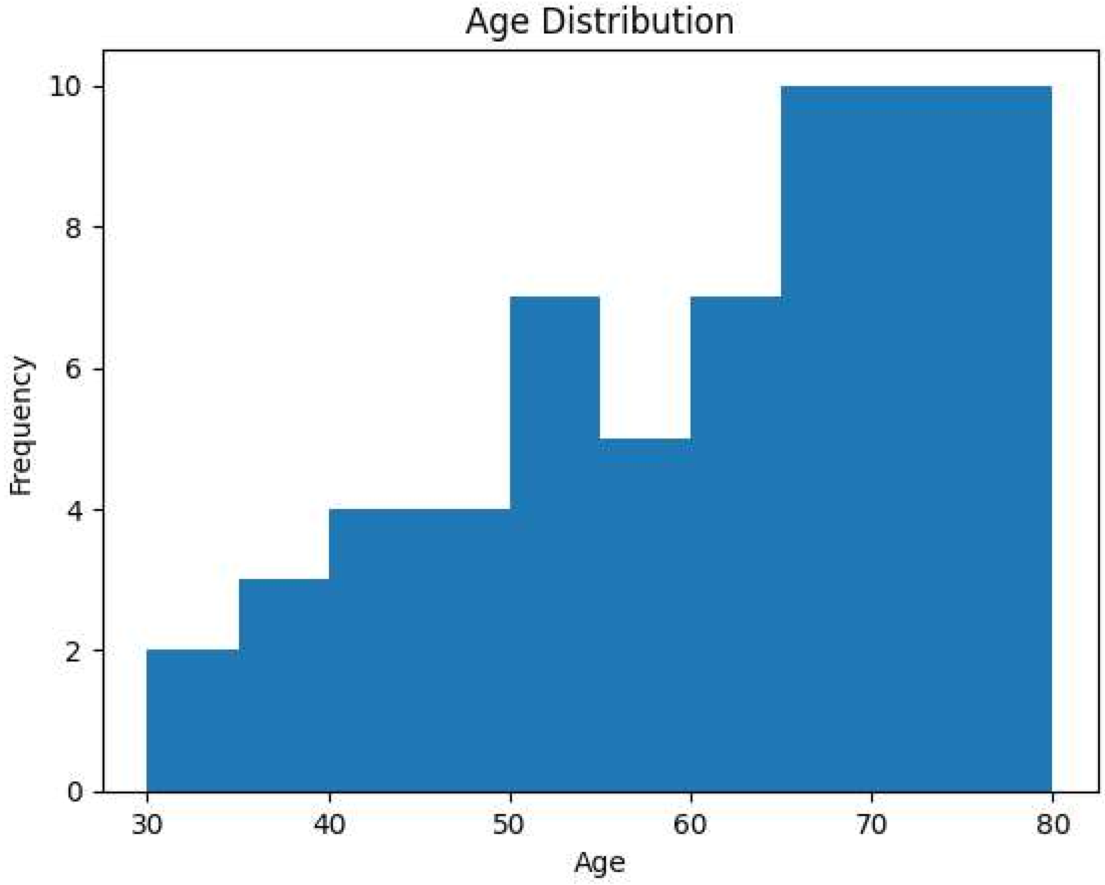
	Religion	Gender	Age	Length of interview (1 =less 30 min, 2 = 30 min to 1 hour,3=1 hour to 2 hours, 4=over two hours)	Village	Nid	A7	B7	C7	D7	E7	F7	G7	H7	I7	Total
0	Skotadi	Male	53	2	Bonriki	B10	2	2	0	0	0	0	0	0	0	4
1	Drepadian	Female	51	1	Bonriki	B11	0	0	0	0	0	0	0	0	0	0
2	Drepadian	Male	60	2	Bonriki	B12	0	0	0	0	0	0	0	0	0	0
3	Thalassic	Male	45	2	Bonriki	B13	0	0	0	0	0	0	0	0	0	0
4	Thalassic	Male	48	2	Bonriki	B14	1	1	0	0	0	0	0	0	0	2
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
57	Drepadian	Female	67	3	Nawerewere	Z10	1	0	0	0	0	0	1	0	0	2
58	Skotadi	Female	49	2	Nawerewere	Z2	0	0	0	0	0	0	0	0	0	0
59	Drepadian	Female	73	4	Nawerewere	Z4	3	1	0	0	1	0	1	0	0	6
60	Skotadi	Female	54	2	Nawerewere	Z5	0	0	0	0	0	0	0	0	0	0
61	Drepadian	Female	30	4	Nawerewere	Z7	0	0	0	0	0	0	0	0	0	0

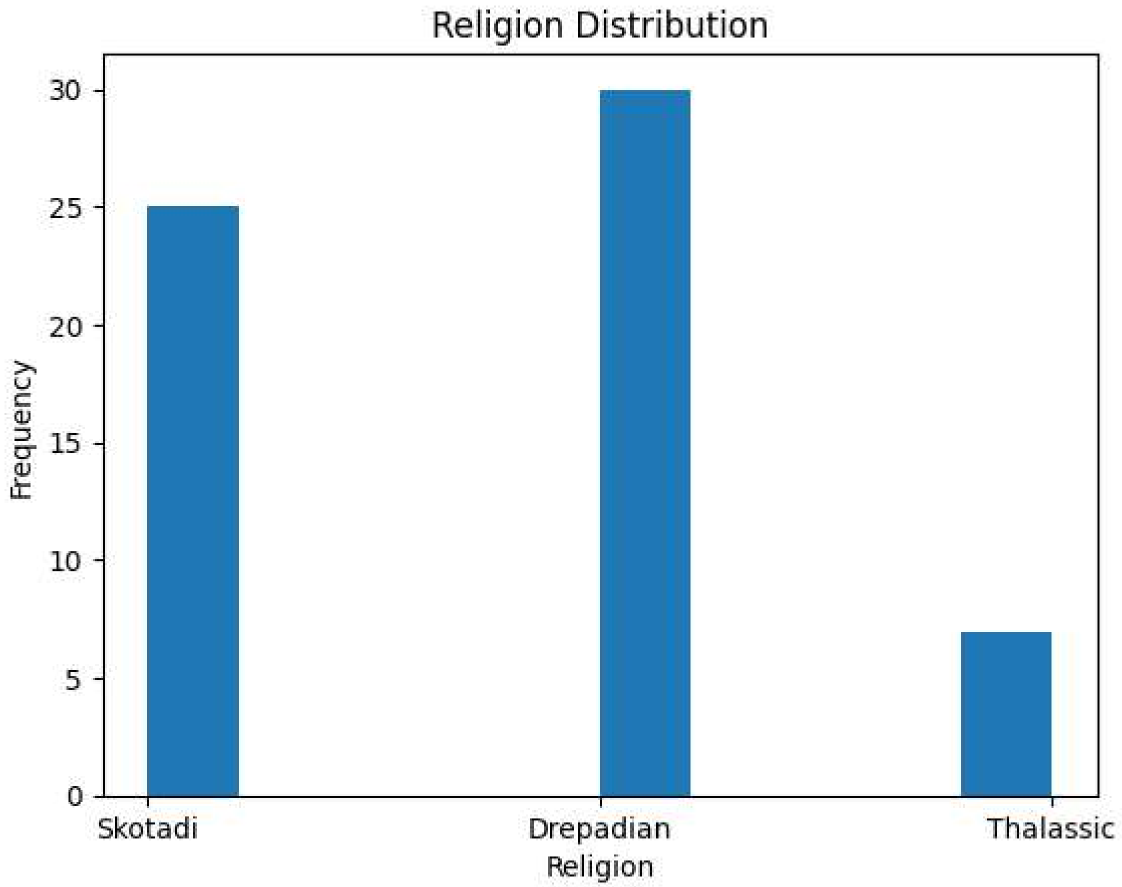
62 rows × 16 columns

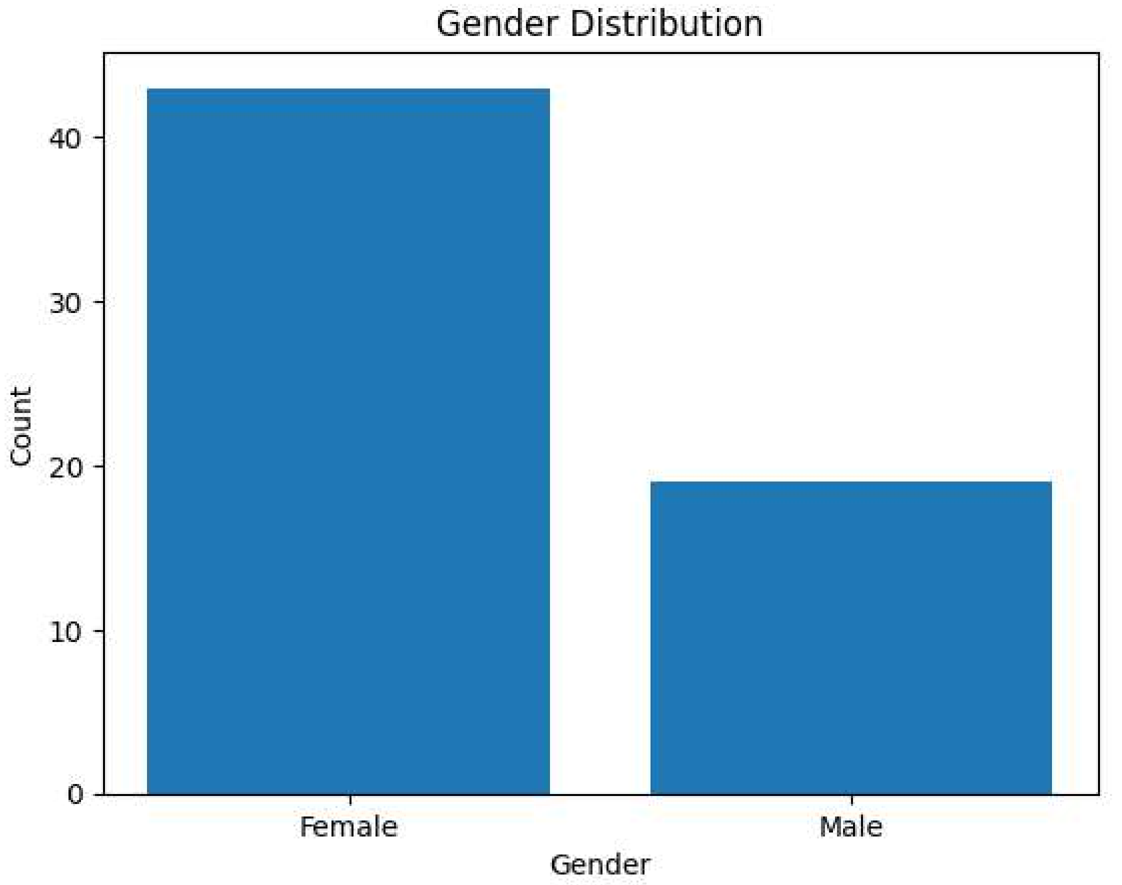
In [8]:

```
# Compute summary statistics for age
age_summary = merged_table[’Age’].describe()
print(age_summary)

# Compute summary statistics for the number of times each form was used
form_counts = merged_table.iloc[:, -10:-1].sum()
form_summary = form_counts.describe()
print(form_summary)

# Plot a histogram of age
plt.hist(merged_table[’Age’])
plt.title(’Age Distribution’)
plt.xlabel(’Age’)
plt.ylabel(’Frequency’)
plt.show()

# Plot a histogram of Religion
plt.hist(merged_table[’Religion’])
plt.title(’Religion Distribution’)
plt.xlabel(’Religion’)
plt.ylabel(’Frequency’)
plt.show()


# Plot a bar chart of gender
gender_counts = merged_table[’Gender’].value_counts()
plt.bar(gender_counts.index, gender_counts.values)
plt.title(’Gender Distribution’)
plt.xlabel(’Gender’)
plt.ylabel(’Count’)
plt.show()
```

```
count    62.000000
mean     60.806452
std      13.048257
min      30.000000
25%      52.000000
50%      64.000000
75%      71.000000
max      80.000000
Name: Age, dtype: float64
count     9.000000
mean     21.777778
std      32.771092
min       1.000000
25%       2.000000
50%       5.000000
75%      32.000000
max      98.000000
dtype: float64
```

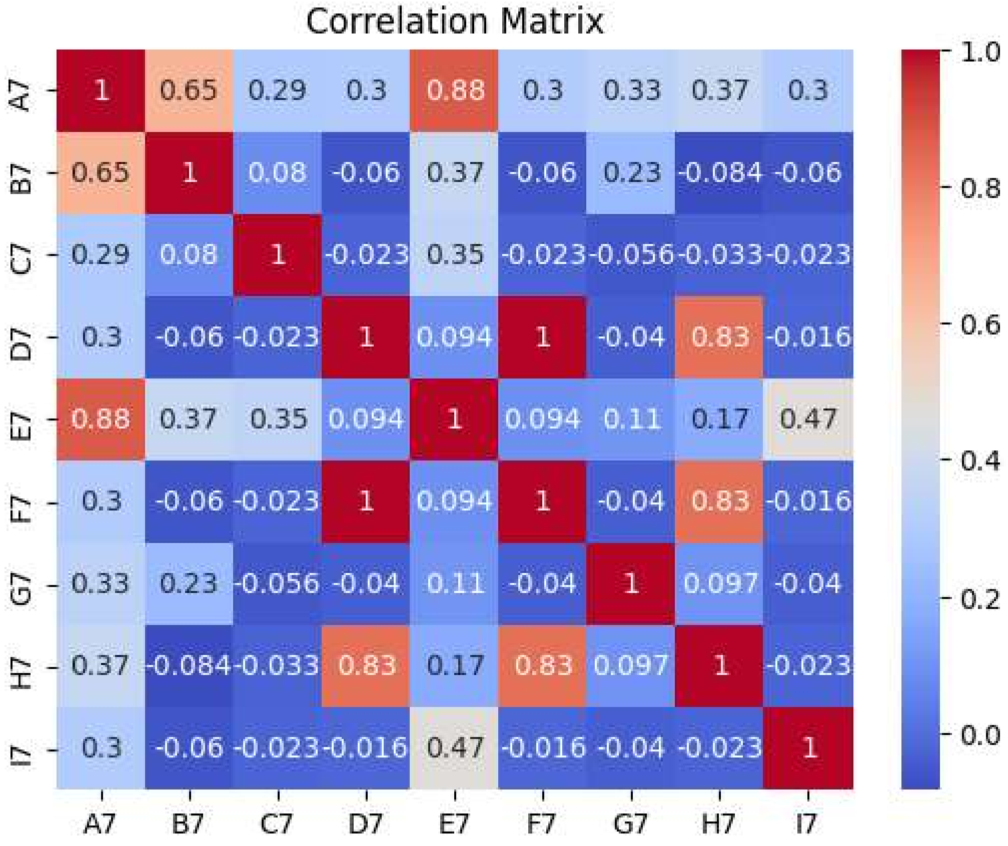






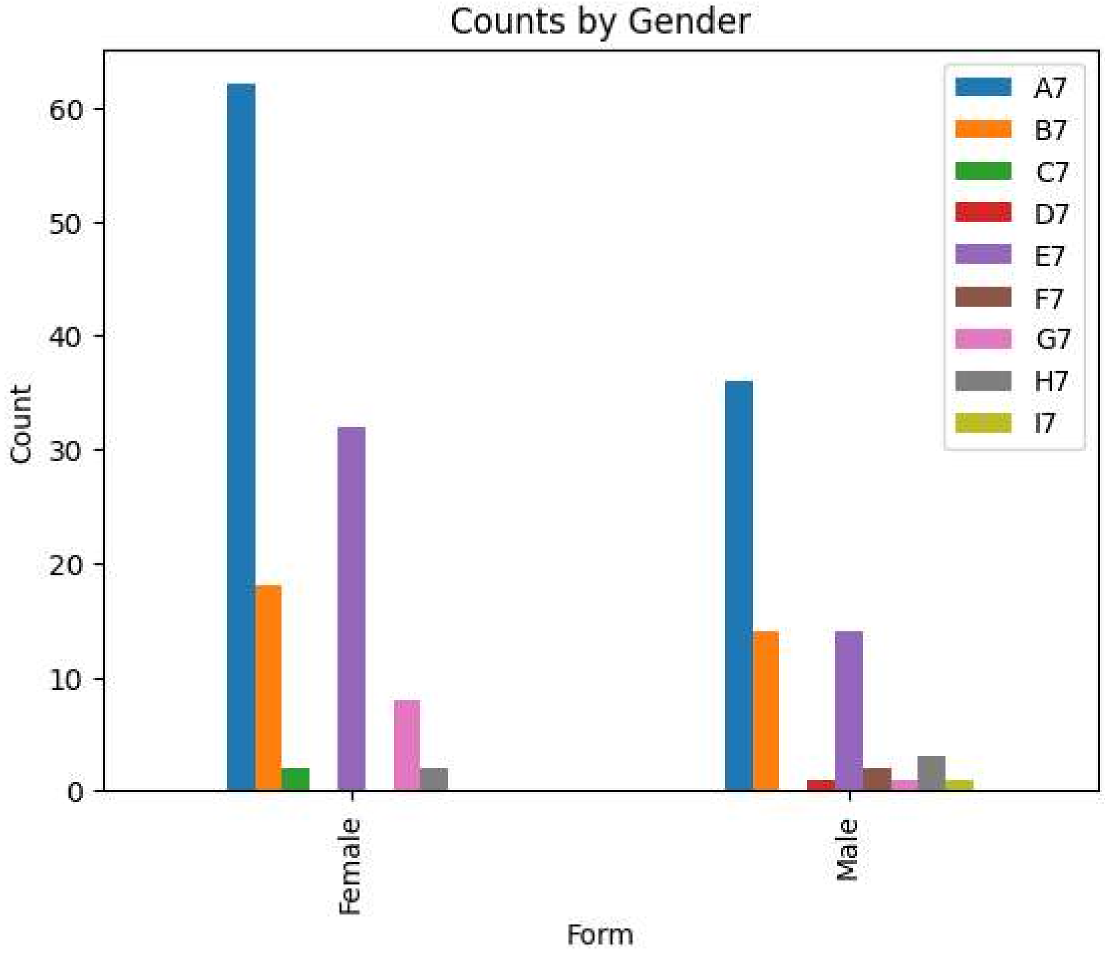
In [9]:

```
# Calculate the correlation matrix
corr_matrix = merged_table.iloc[:, -10:-1].corr()

# Plot a heatmap of the correlation matrix
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm')
plt.title('Correlation Matrix')
plt.show()
```



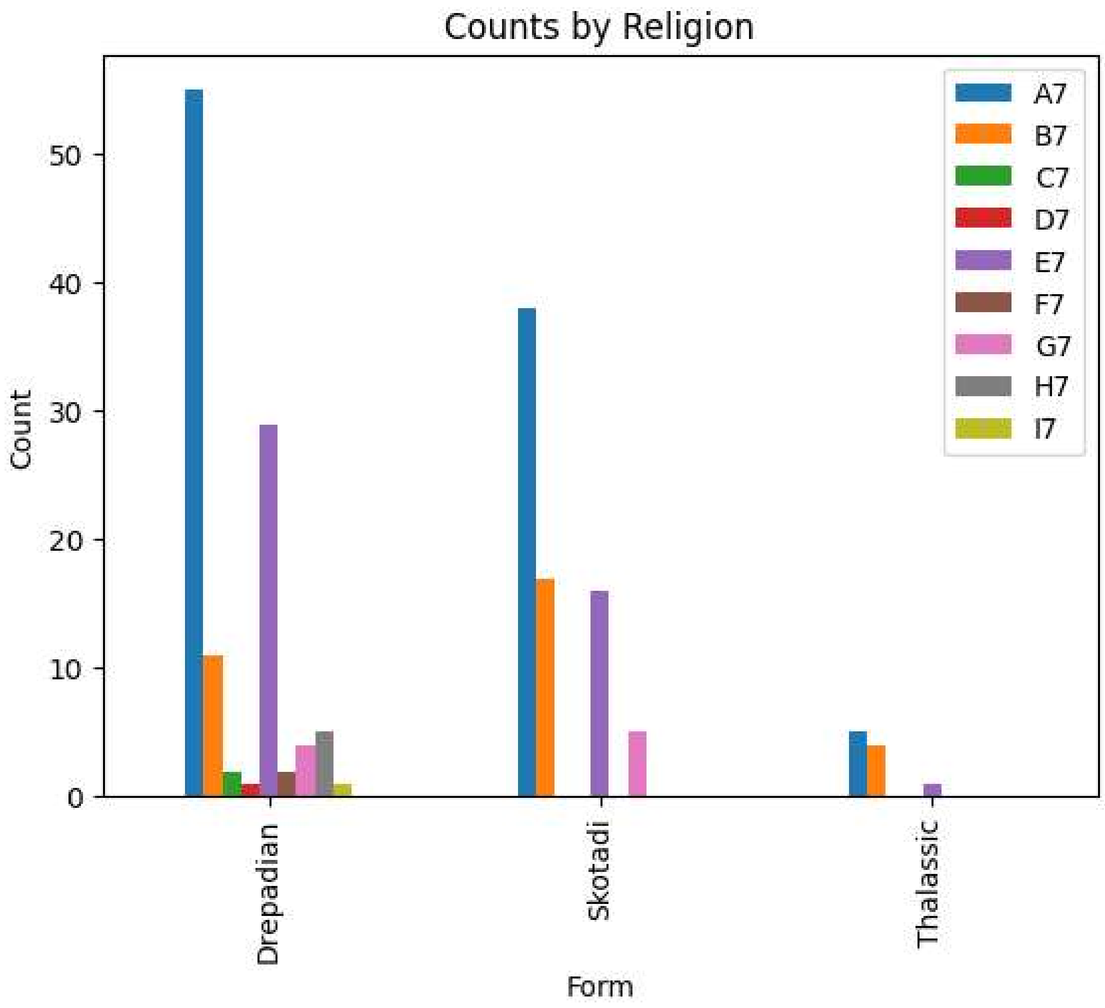
In [10]:

```
# Group by gender and calculate the sum of counts for each form
gender_counts = merged_table.groupby('Gender')[['A7', 'B7', 'C7', 'D7', 'E7', 'F7', 'G7', 'H7',

# Plot a bar chart of the counts by gender
gender_counts.plot(kind='bar')
plt.title('Counts by Gender')
plt.xlabel('Form')
plt.ylabel('Count')
plt.show()
```



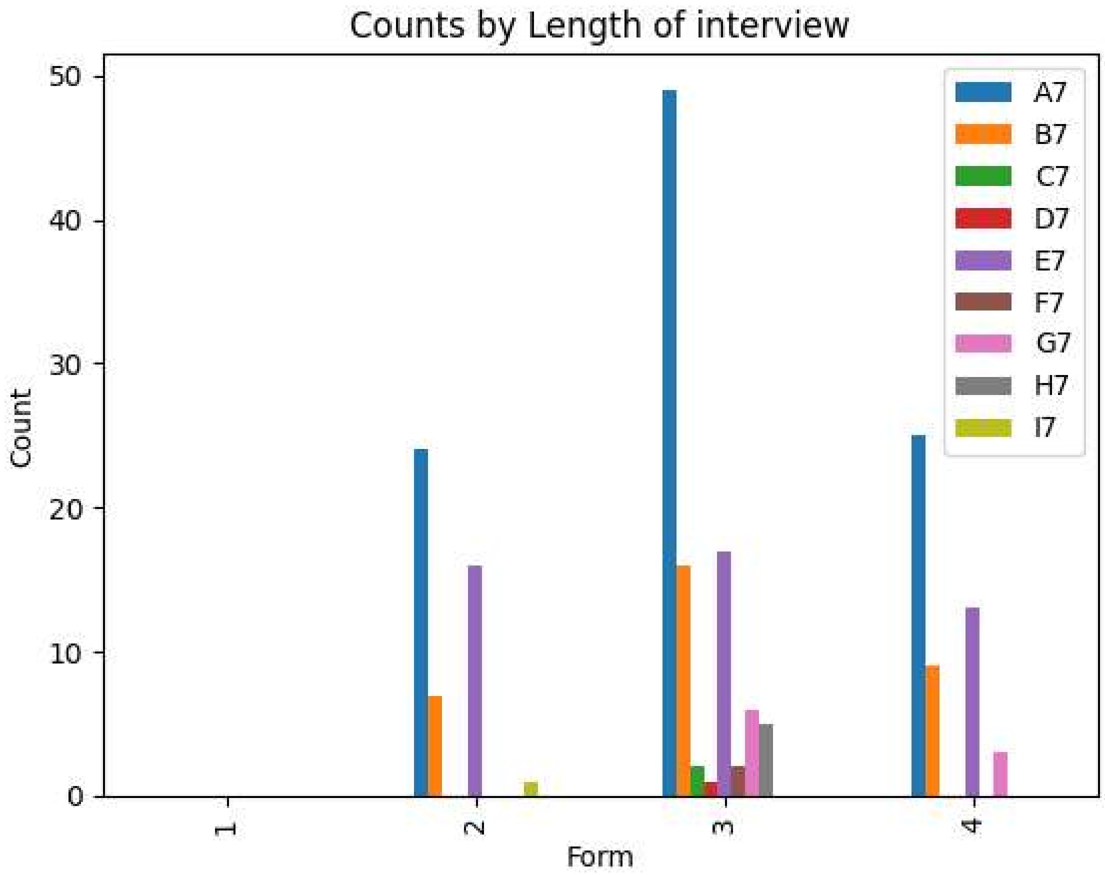
In [11]:

```
# Group by Religion and calculate the sum of counts for each form
Religion_counts = merged_table.groupby('Religion')[['A7', 'B7', 'C7', 'D7', 'E7', 'F7', 'G7', '

# Plot a bar chart of the counts by Religion
Religion_counts.plot(kind='bar')
plt.title('Counts by Religion')
plt.xlabel('Form')
plt.ylabel('Count')
plt.show()
```



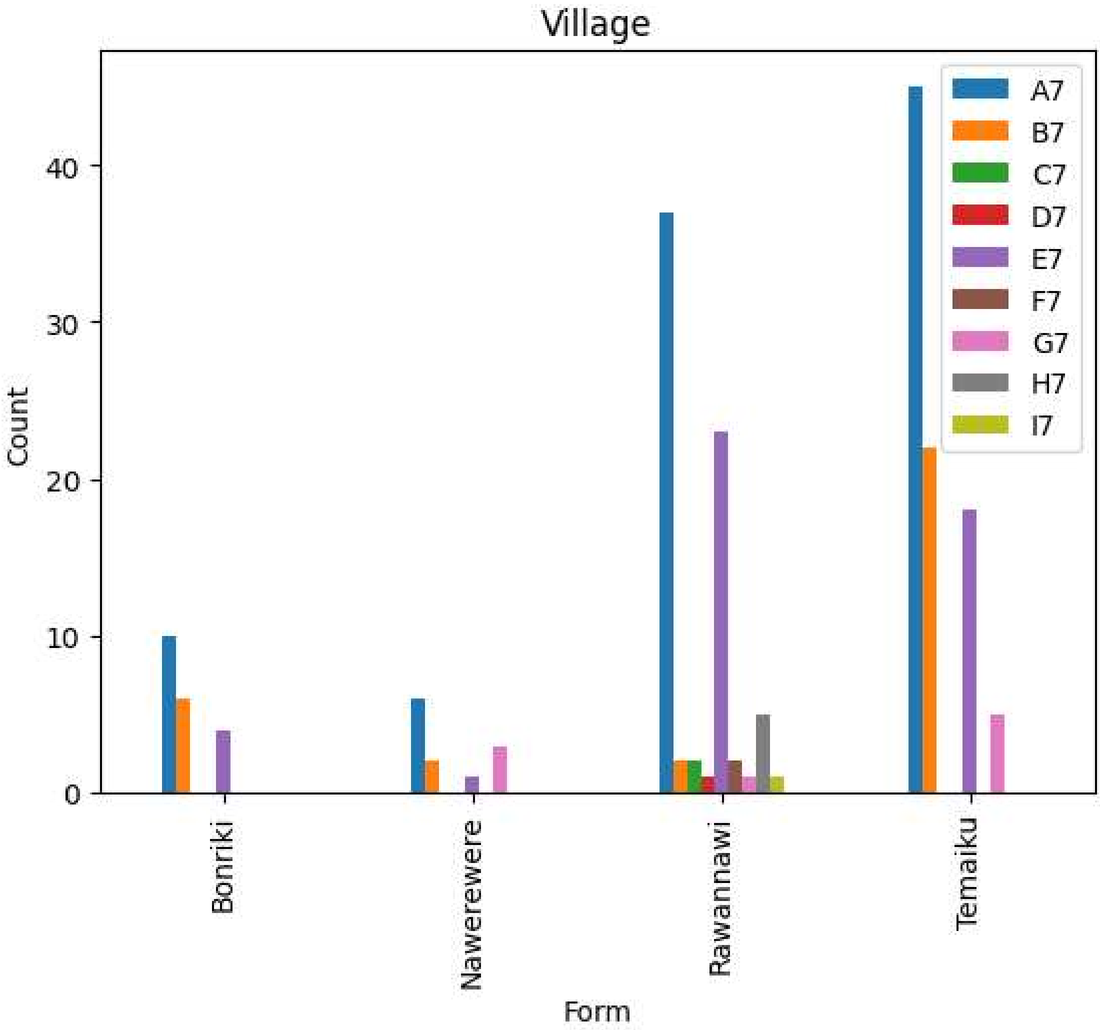
In [12]:

```
# Group by Length of interview and calculate the sum of counts for each form
Length_counts = merged_table.groupby('Length of interview ( 1 =less 30 min, 2 = 30 min to 1 hou

# Plot a bar chart of the counts by Length of interview
Length_counts.plot(kind='bar')
plt.title('Counts by Length of interview')
plt.xlabel('Form')
plt.ylabel('Count')
plt.show()
```



In [13]:

```
# Group by Length of interview and calculate the sum of counts for each form
Village_counts = merged_table.groupby('Village')[['A7', 'B7', 'C7', 'D7', 'E7', 'F7', 'G7', 'H7

# Plot a bar chart of the counts by Length of interview
Village_counts.plot(kind='bar')
plt.title('Village')
plt.xlabel('Form')
plt.ylabel('Count')
plt.show()
```


Village
Count
0
10
20
30
40
A7
B7
C7
D7
E7
F7
G7
H7
I7
Bonriki
Nawerewere
Rawannawi
Temaiku
Form

In [ ]: