CENSUS 14 PROJECT REPORT

Introduction

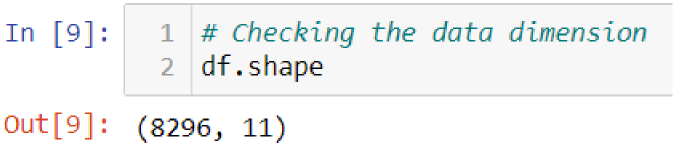
The town from the census is a modestly sized one sandwiched between two much larger cities that it is connected to by motorways. The town does not have a university, but students do live in the town and commute to the nearby cities.

Objectives

- Clean the census data
- Analyze the data
- Visualize the data and
- Take decision as part of a Local Government team on what to do with an unoccupied plot of land and what to invest in.

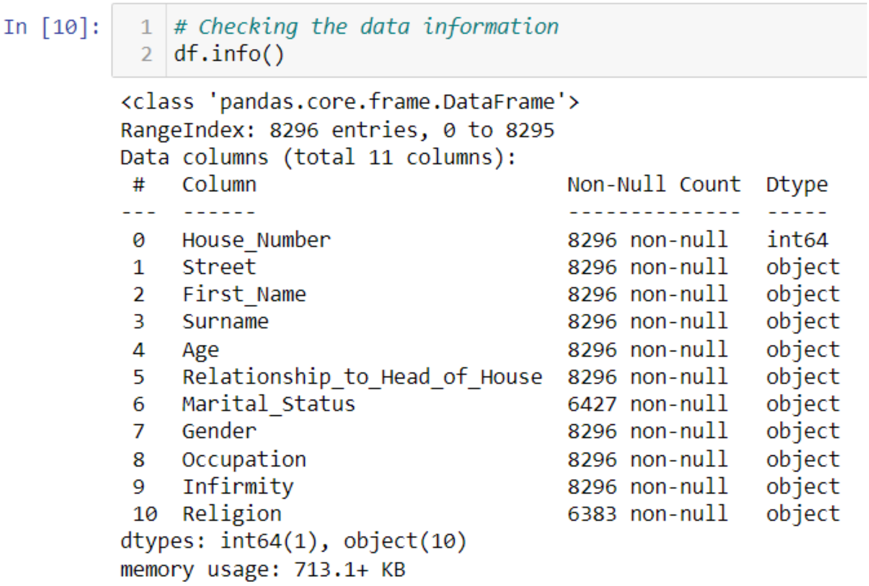
Data Preprocessing and Exploration

Data Shape: The dataset contains 8296 rows and 11 columns. The shape of the dataset was checked using *df.shape* method.



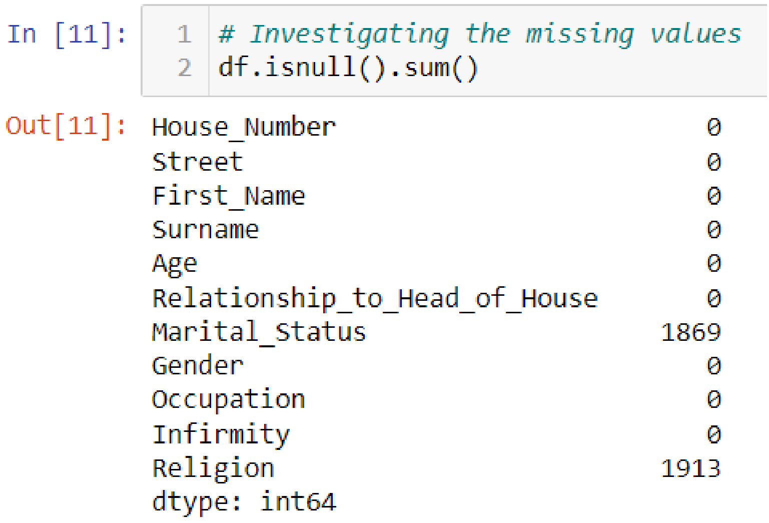
```
In [9]:  # Checking the data dimension
          df.shape

Out[9]:  (8296, 11)
```

Data Information: In a bid to understand the data more, *df.info()* method was used to extract more information from the census data. Out of the 11 columns, 10 are of the object data type and 1 is of the integer data type. The Marital Status and Religion features had missing values which were addressed. Additionally, the Age feature was initially represented as a string and was converted to an integer since age is typically expressed in whole numbers.



```
In [10]:  # Checking the data information
          df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8296 entries, 0 to 8295
Data columns (total 11 columns):
 #   Column                         Non-Null Count  Dtype 
---  ------                         --------------  ----- 
 0   House_Number                   8296 non-null   int64 
 1   Street                         8296 non-null   object
 2   First_Name                     8296 non-null   object
 3   Surname                        8296 non-null   object
 4   Age                            8296 non-null   object
 5   Relationship_to_Head_of_House  8296 non-null   object
 6   Marital_Status                 6427 non-null   object
 7   Gender                         8296 non-null   object
 8   Occupation                     8296 non-null   object
 9   Infirmity                      8296 non-null   object
 10  Religion                       6383 non-null   object
dtypes: int64(1), object(10)
memory usage: 713.1+ KB
```

Missing Values: df.isnull().sum() method was used to check the missing values in the census data. The Marital Status feature had 1869 missing values, while the Religion feature had 1913 missing values.



```
In [11]:  # Investigating the missing values
          df.isnull().sum()

Out[11]:  House_Number                     0
          Street                           0
          First_Name                       0
          Surname                          0
          Age                              0
          Relationship_to_Head_of_House    0
          Marital_Status                1869
          Gender                           0
          Occupation                       0
          Infirmity                        0
          Religion                      1913
          dtype: int64
```

Processing Age Feature

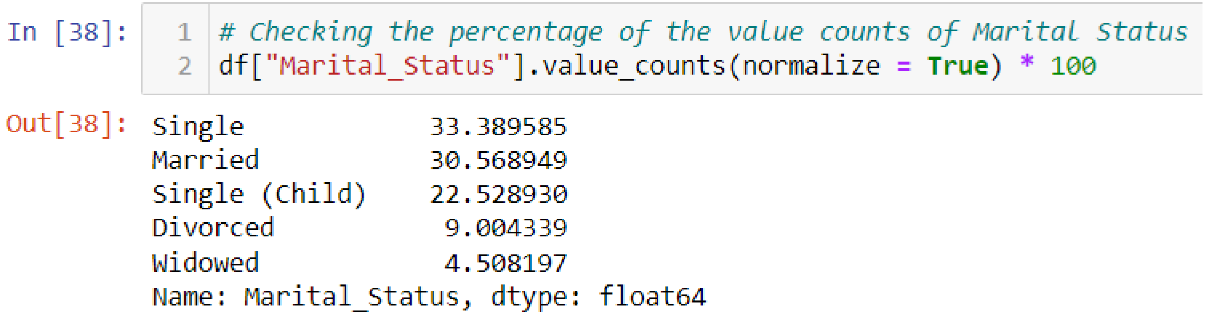
The Age feature consists of empty strings, strings, and floating point numbers, which was converted to integers. Based on the value counts, it was observed that the dominant age in the population was 37. There was no missing value in the Age feature. However, the Age Feature contains an empty string. In a bid to fill the missing data point, other members in the household were examined and explored whether their ages could guide in imputing a value for the empty string in Age. Further investigation was carried out whether the median or mean age of the cousins and heads of households in the population could be utilized to fill in the missing values. According to the results obtained, the average age of heads of households in the population was 49, while the average age of their cousins was 41. The median ages of the heads and cousins were 47 and 42, respectively. As the median is not influenced by outliers, the difference in the median ages (which is 5) was used as the age difference between the individual with the missing age and his Head of House.

Processing Marital Status Feature

After conducting a value count analysis, it was observed that singles were the most represented group in the population, while widows were the least represented. Furthermore, no empty string was found in the feature. However, the feature contained 1,869 missing values. After further analysis, it became clear that these missing values were exclusively for individuals under 18 years of age, who are classified as children in the United Kingdom. These values were imputed with the label "Single (Child)."

Reference:
https://www.gov.uk/guidance/case-management-guidance/definitions#:~:text=We%20define%20a%20child%20as,legislation%20in%20England%20and%20Wales.

After checking the percentage of the value counts of Marital Status, it was observed that 33.4% of the population is comprised of individuals who are unmarried, 30.6% are married, 22.5% are unmarried and under the age of 18, 9% are divorced, and 4.5% are widowed.



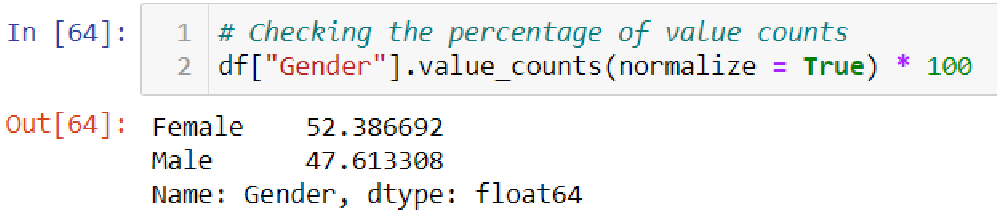
```
In [38]:  # Checking the percentage of the value counts of Marital Status
           df["Marital_Status"].value_counts(normalize = True) * 100
```

```
Out[38]: Single            33.389585
         Married           30.568949
         Single (Child)    22.528930
         Divorced           9.004339
         Widowed            4.508197
         Name: Marital_Status, dtype: float64
```

Based on the investigation carried out, it appears that two individuals who are 18 years of age and identified as students (not university students) are also widowed. Moreover, one of them was identified as a grandson to the head of the household. This questions the accuracy of their recorded marital status. Consequently, their marital status was updated to "Single" given that they are no longer minors.

Processing Gender Feature

The Gender feature suggests that females are about 52% of the population while the males are about 48% of the population. No missing value or empty string was present in the feature.



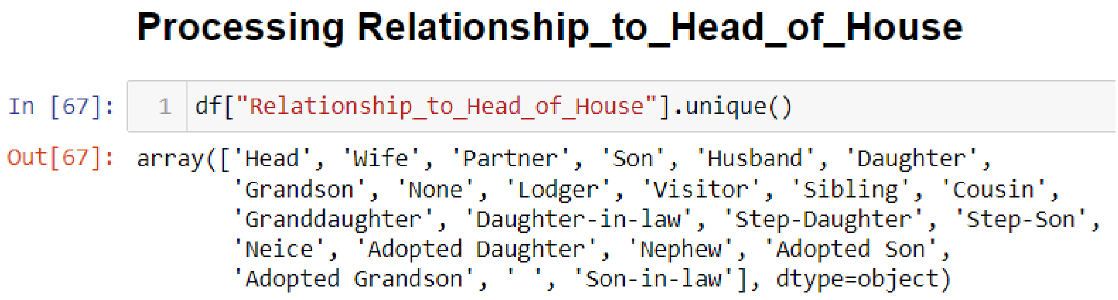
```
In [64]:  # Checking the percentage of value counts
           df["Gender"].value_counts(normalize = True) * 100
```

```
Out[64]: Female    52.386692
         Male      47.613308
         Name: Gender, dtype: float64
```

Processing Relationship_to_Head_of_House Feature

During the analysis, it was discovered that "Neice" was present as one of the unique values under the feature "Relationship_to_Head_of_House", which is a misspelling of "Niece." This error has been corrected.

Furthermore, an empty string was identified as one of the unique values in "Relationship_to_Head_of_House," which has been replaced.



Processing Relationship_to_Head_of_House

```
In [67]:  df["Relationship_to_Head_of_House"].unique()
```

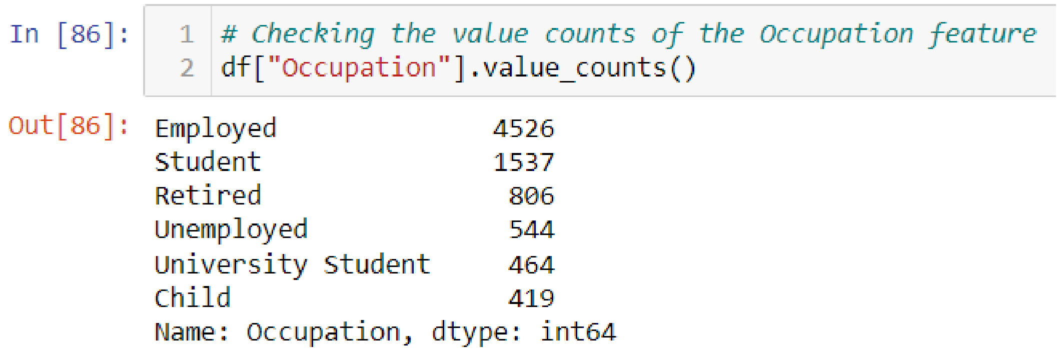
```
Out[67]: array(['Head', 'Wife', 'Partner', 'Son', 'Husband', 'Daughter',
                'Grandson', 'None', 'Lodger', 'Visitor', 'Sibling', 'Cousin',
                'Granddaughter', 'Daughter-in-law', 'Step-Daughter', 'Step-Son',
                'Neice', 'Adopted Daughter', 'Nephew', 'Adopted Son',
                'Adopted Grandson', ' ', 'Son-in-law'], dtype=object)
```

According to the value count percentage, the population is predominantly made up of Head of Households (39%), while the Son-in-law (0.012%) is the least represented relationship category.

Based on an analysis conducted, it was discovered that there are 23 records of individuals who were listed as the Head of House with an age less than 19. This is contradictory to the information provided with the dataset, which stated that only individuals above 18 years can be listed as the Head of House. Upon further investigation, it was found that most of these records belong to high school students, indicating that they were likely mistakenly listed as the Head of House. As a result, these records were considered invalid.

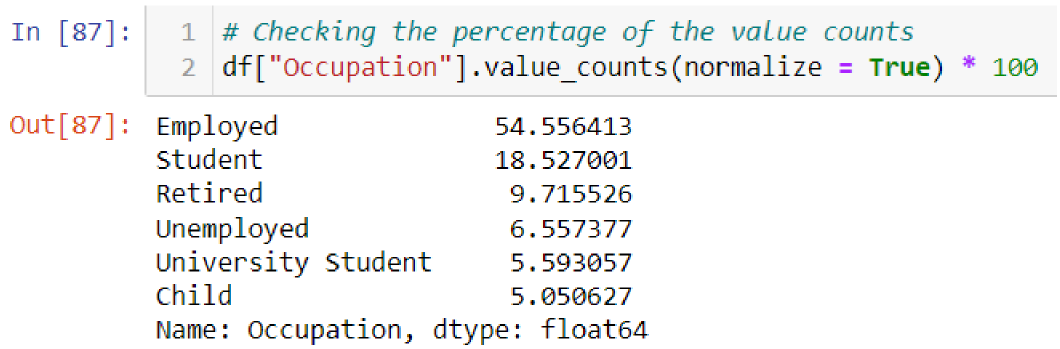
Processing Occupation Feature

Based on the value counts, 1108 unique values were identified in the Occupation feature. To simplify the analysis, all employed individuals (4,526) were grouped under the "employed" category while the retired individuals (806) were grouped under the "retired" category.



```
In [86]:  # Checking the value counts of the Occupation feature
          df["Occupation"].value_counts()

Out[86]:  Employed              4526
          Student               1537
          Retired                806
          Unemployed             544
          University Student     464
          Child                  419
          Name: Occupation, dtype: int64
```

Based on the value counts percentage, the employed is about 55% of the population. This figure represents the proportion of the population that is contributing to the economy through their labor. A high level of employment can be seen as a positive sign for the economy.



```
In [87]:  # Checking the percentage of the value counts
          df["Occupation"].value_counts(normalize = True) * 100

Out[87]:  Employed              54.556413
          Student               18.527001
          Retired                9.715526
          Unemployed             6.557377
          University Student     5.593057
          Child                  5.050627
          Name: Occupation, dtype: float64
```

It is also noteworthy that there are 124 individuals that were above the popular retirement age of 65 but are employed. This is not against the current UK law. The UK law states that "You can usually work for as long as you want to. ‘Default retirement age’ (a forced retirement age of 65) no longer exists". *Reference:* https://www.gov.uk/working-retirement-pension-age.

Following an investigation, 41 individuals were older than 65 and identified as unemployed. It would be more appropriate if they are grouped under the retired catogory. Hence the individuals were moved to the retired category.

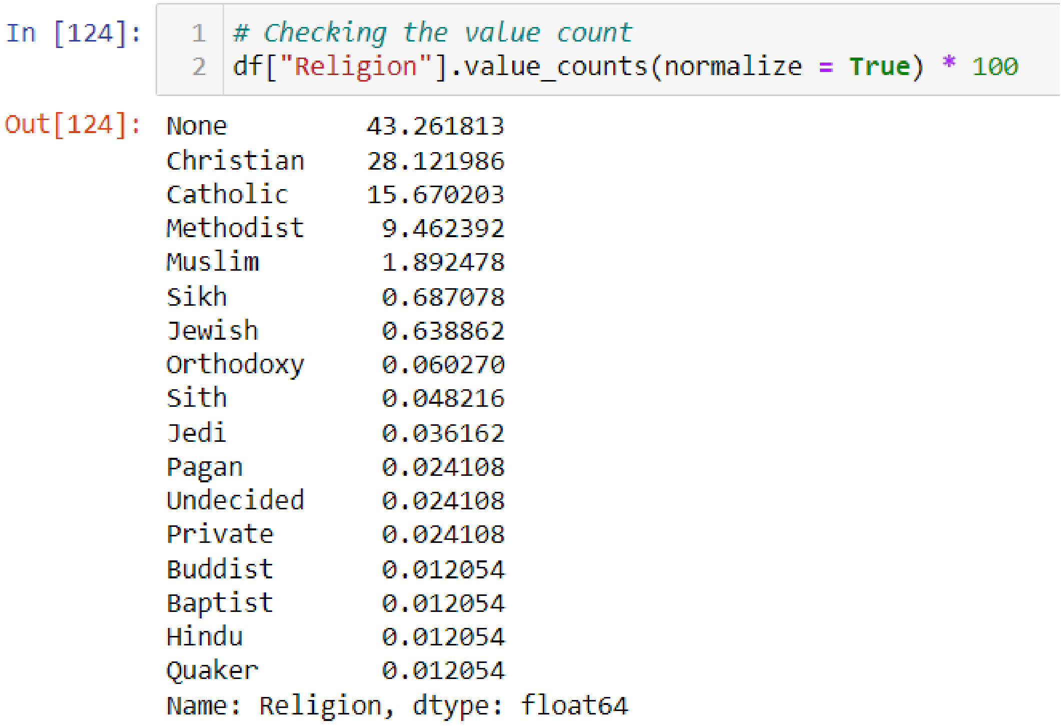
Processing Religion Feature

Based on the value count percentage, 43% of the population reported having no religion. Christianity was the most common religion among the population, with approximately 28% of individuals identifying as such. The Catholic faith followed with approximately 16%, followed by Methodism at around 10%, and Islam with approximately 2%. All other religions constituted less than 1% of the population each.

An empty string was present in the feature but was imputed using the religion of the Head of the Household which may not necessarily be accurate.

Upon analysis, it was discovered that there were 1,913 missing values in the religion feature. Further investigation revealed that out of the total of 1,913 individuals with unspecified religions, 1,870 were children. To fill in the missing values, an assumption was made by assigning the religion of the head of the household to the missing religions of the children who bore the same surname as the head of the household. However, it is important to note that this assumption may not be applicable.

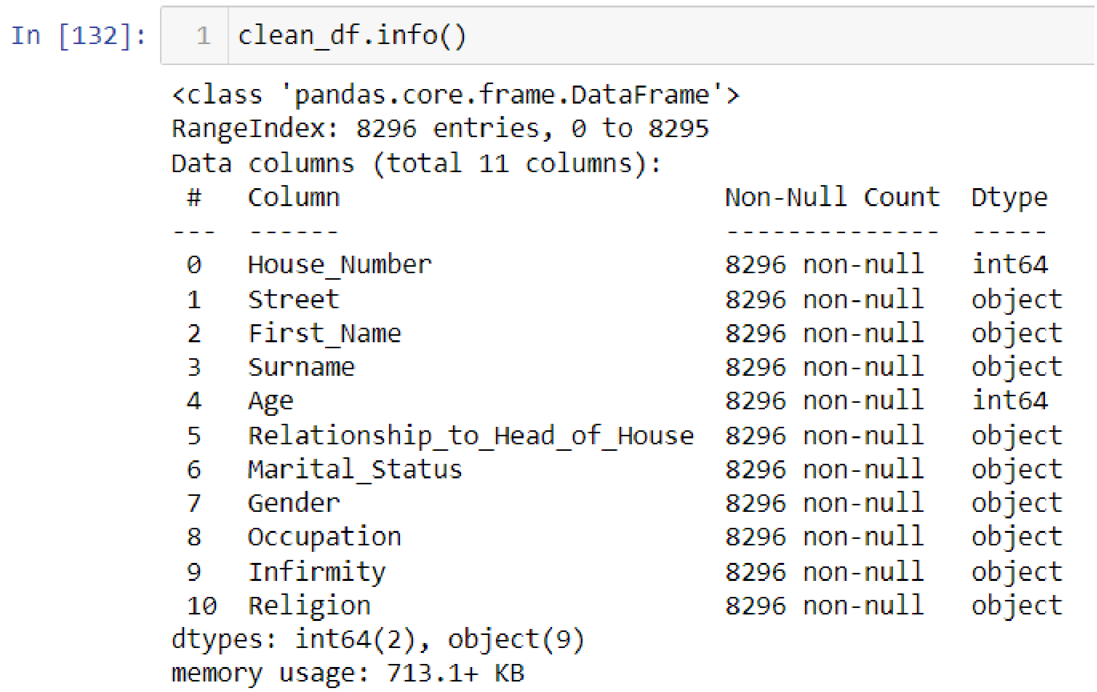
It is important to recognize that in the UK, parents have the right to raise their children according to their own religious beliefs. Legal frameworks are in place to protect these rights. However, it is prohibited to coerce a child into following a specific religion or restrict them from exploring and practicing other beliefs, as stipulated by The Human Rights Act 1998.



```
In [124]:
# Checking the value count
df["Religion"].value_counts(normalize = True) * 100
```

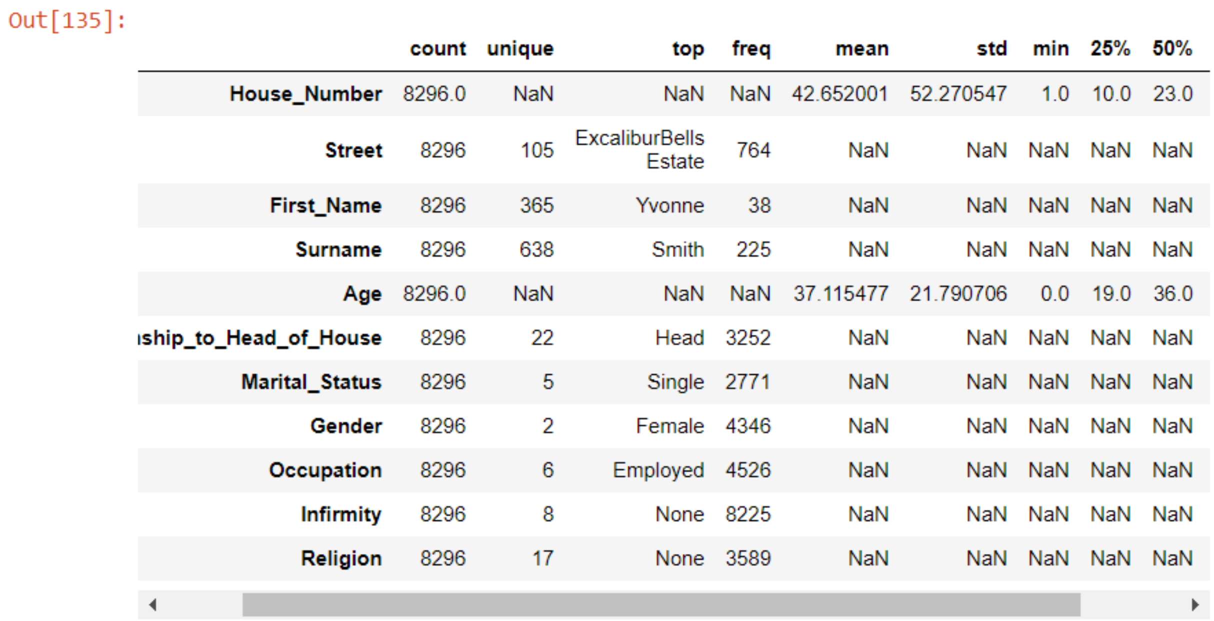
```
Out[124]: None          43.261813
          Christian     28.121986
          Catholic      15.670203
          Methodist      9.462392
          Muslim         1.892478
          Sikh           0.687078
          Jewish         0.638862
          Orthodoxy      0.060270
          Sith           0.048216
          Jedi           0.036162
          Pagan          0.024108
          Undecided      0.024108
          Private        0.024108
          Buddist        0.012054
          Baptist        0.012054
          Hindu          0.012054
          Quaker         0.012054
          Name: Religion, dtype: float64
```

After the dataset has been fully processed, the clean dataset was saved to a new CSV file for visualization and further analysis. Below is the data information of the cleaned census data:



```
In [132]:  clean_df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8296 entries, 0 to 8295
Data columns (total 11 columns):
 #   Column                        Non-Null Count  Dtype 
---  ------                        --------------  ----- 
 0   House_Number                  8296 non-null   int64 
 1   Street                        8296 non-null   object
 2   First_Name                    8296 non-null   object
 3   Surname                       8296 non-null   object
 4   Age                           8296 non-null   int64 
 5   Relationship_to_Head_of_House 8296 non-null   object
 6   Marital_Status                8296 non-null   object
 7   Gender                        8296 non-null   object
 8   Occupation                    8296 non-null   object
 9   Infirmity                     8296 non-null   object
 10  Religion                      8296 non-null   object
dtypes: int64(2), object(9)
memory usage: 713.1+ KB
```

Based on this information, it's evident that there are no more missing values and the age data type has been changed from object (string) data type to integer. Below is the descriptive statistics of the processed census data:



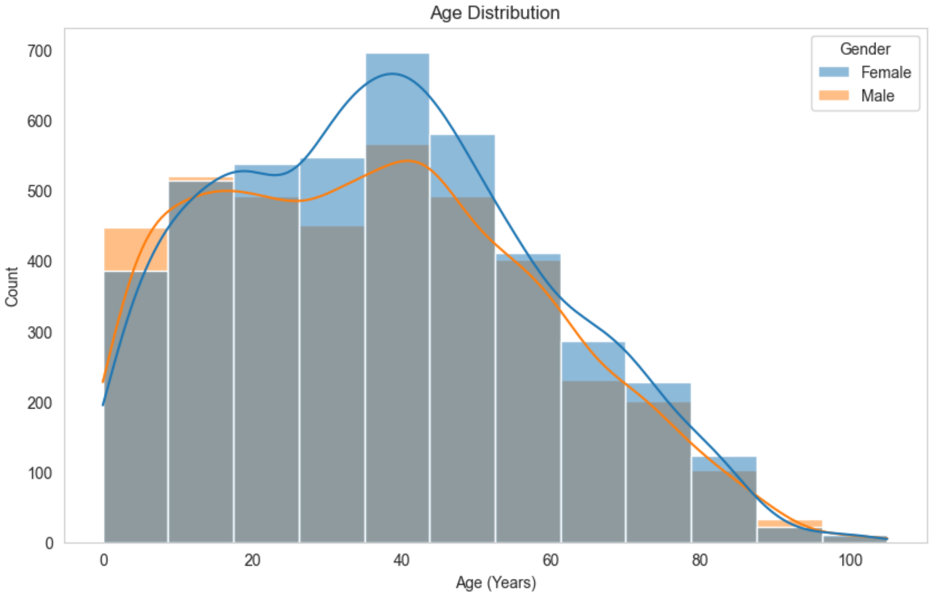
Out[135]:

	count	unique	top	freq	mean	std	min	25%	50%
House_Number	8296.0	NaN	NaN	NaN	42.652001	52.270547	1.0	10.0	23.0
Street	8296	105	ExcaliburBells Estate	764	NaN	NaN	NaN	NaN	NaN
First_Name	8296	365	Yvonne	38	NaN	NaN	NaN	NaN	NaN
Surname	8296	638	Smith	225	NaN	NaN	NaN	NaN	NaN
Age	8296.0	NaN	NaN	NaN	37.115477	21.790706	0.0	19.0	36.0
ship_to_Head_of_House	8296	22	Head	3252	NaN	NaN	NaN	NaN	NaN
Marital_Status	8296	5	Single	2771	NaN	NaN	NaN	NaN	NaN
Gender	8296	2	Female	4346	NaN	NaN	NaN	NaN	NaN
Occupation	8296	6	Employed	4526	NaN	NaN	NaN	NaN	NaN
Infirmity	8296	8	None	8225	NaN	NaN	NaN	NaN	NaN
Religion	8296	17	None	3589	NaN	NaN	NaN	NaN	NaN

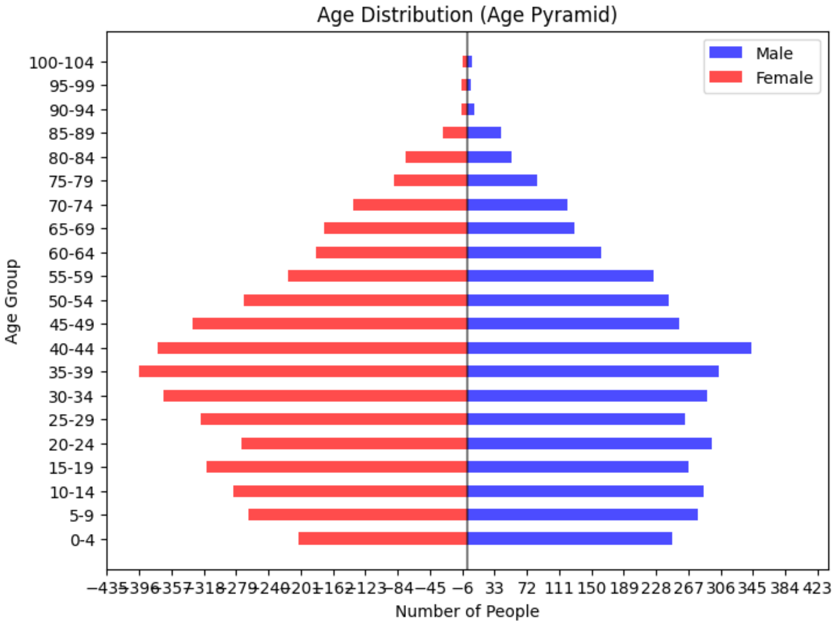
The descriptive statistics above indicate that the population comprises mainly Heads, Singles, Females, and Employed Individuals. Moreover, a substantial proportion of the population reported having no infirmities, and a higher percentage identified as having no religion.

Data Visualization

Visualising Age Feature



Based on the histogram plot displayed above, it is evident that the male population, female population, and the overall population have experienced a decline in the recent years.



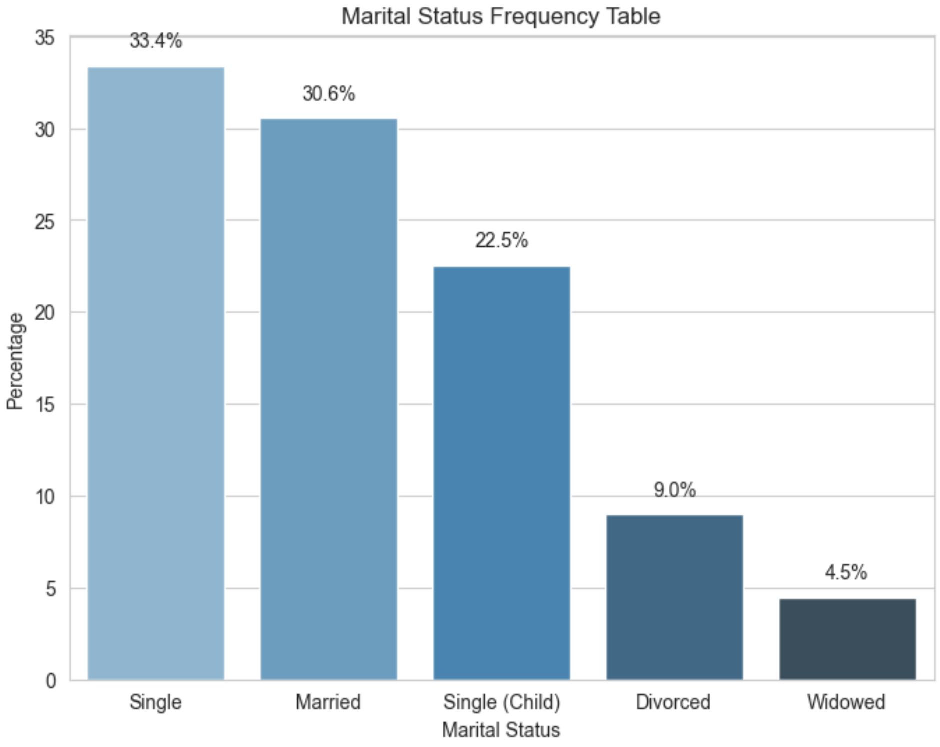
Observing the age pyramid plot above, it is notable that the widest base of the pyramid lies within the age ranges of 35-39 and 40-44. However, the base has reduced from the age range of 30-34 to 0-4, implying that birth rates have declined in recent years.

The fact that the base of the pyramid has narrowed from age range 30-34 and below indicates that there are fewer younger individuals in the population than there were in the past. This could be due to a variety of factors, such as changes in fertility rates, increased access to contraception, or changes in societal norms around family planning.

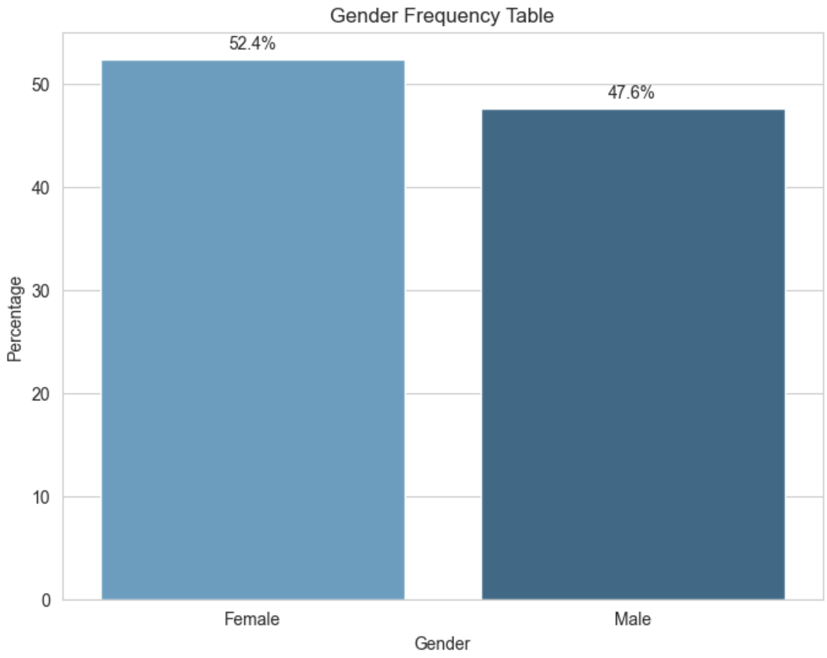
The fact that the widest base of the pyramid is in the age range of 35-39 and 40-44 suggests that the population is aging, as these individuals are moving up the pyramid and entering older age groups. This could have implications for issues such as healthcare, retirement, and social security, as there may be a larger number of older individuals in the population who require these services.

In addition, the fact that the base has reduced to the 0-4 age range suggests that there may be a smaller cohort of children in the population. This could have implications for issues such as education and childcare, as there may be a smaller pool of young individuals to support these systems.

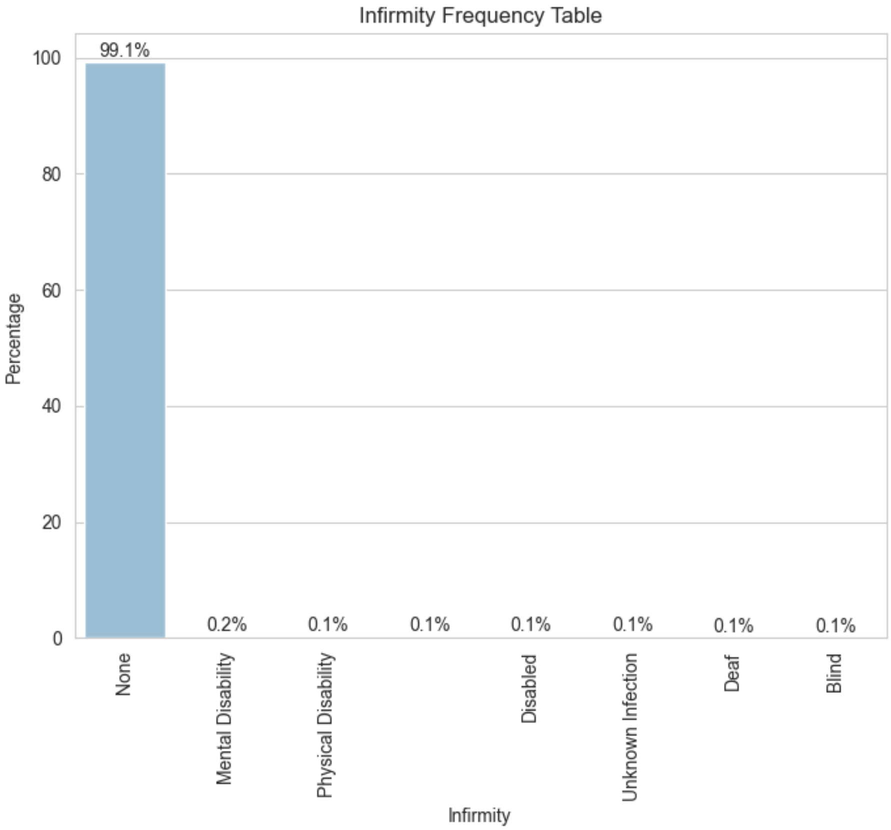
Visualising Marital Status Feature



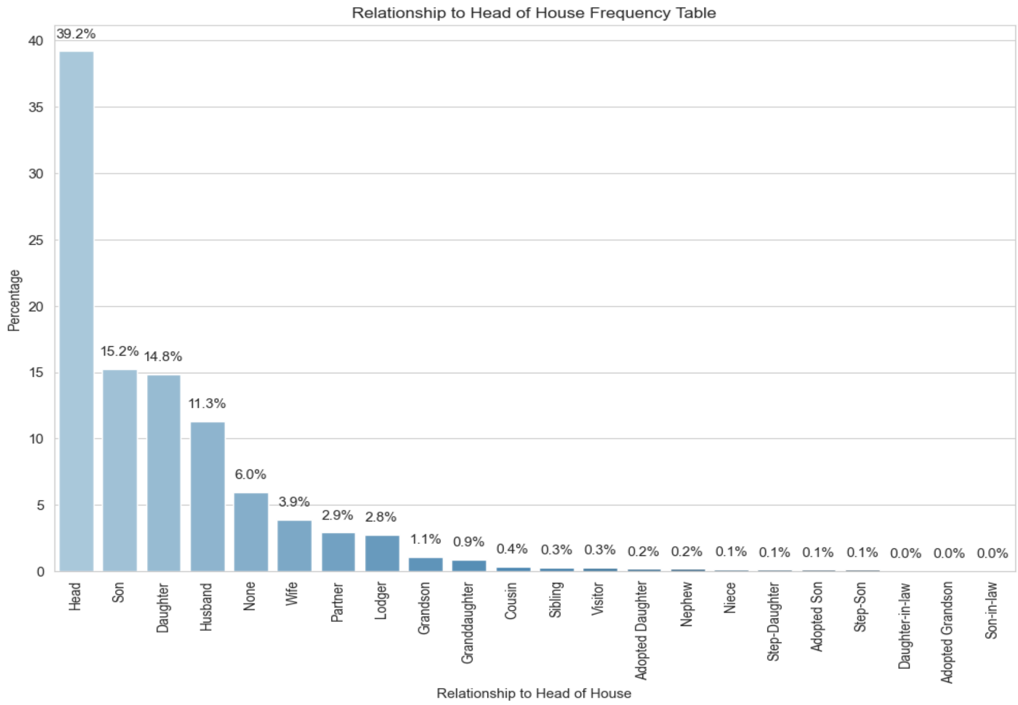
Visualising Gender Feature



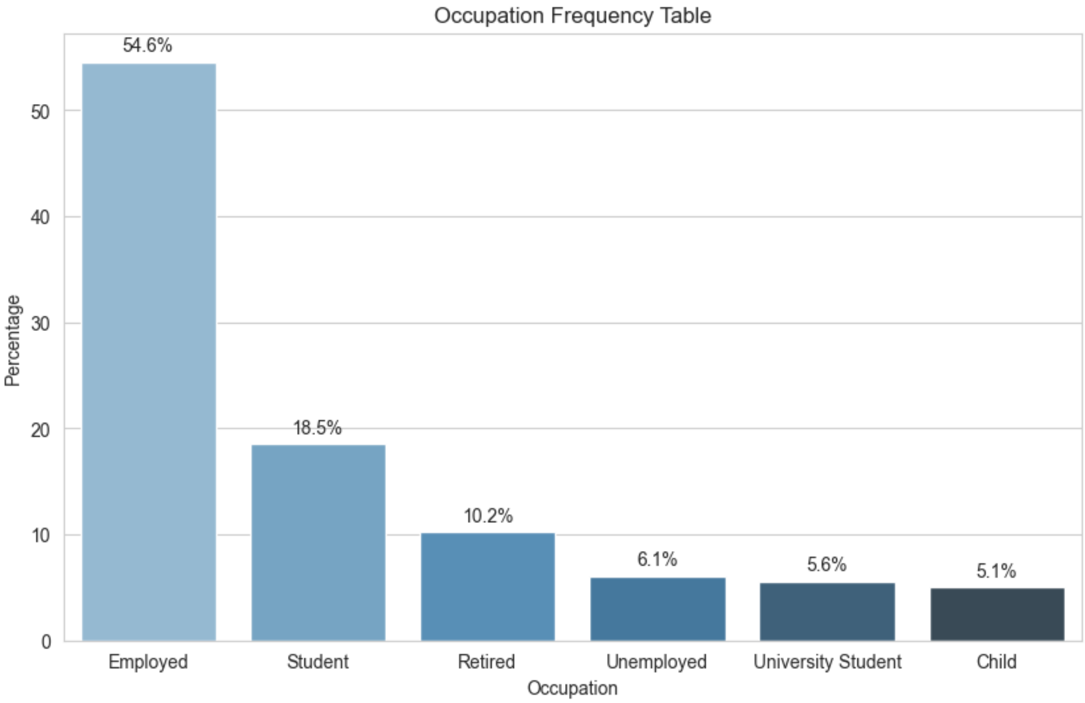
Visualising Infirmity Feature



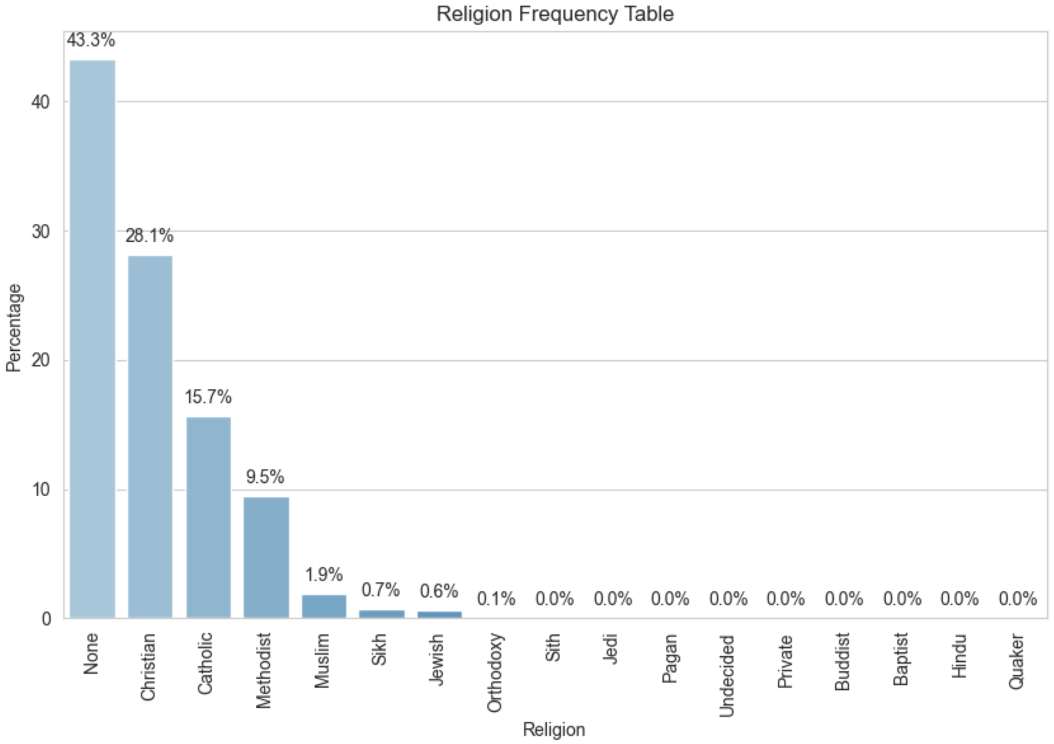
Visualising Relationship to Head of House Feature



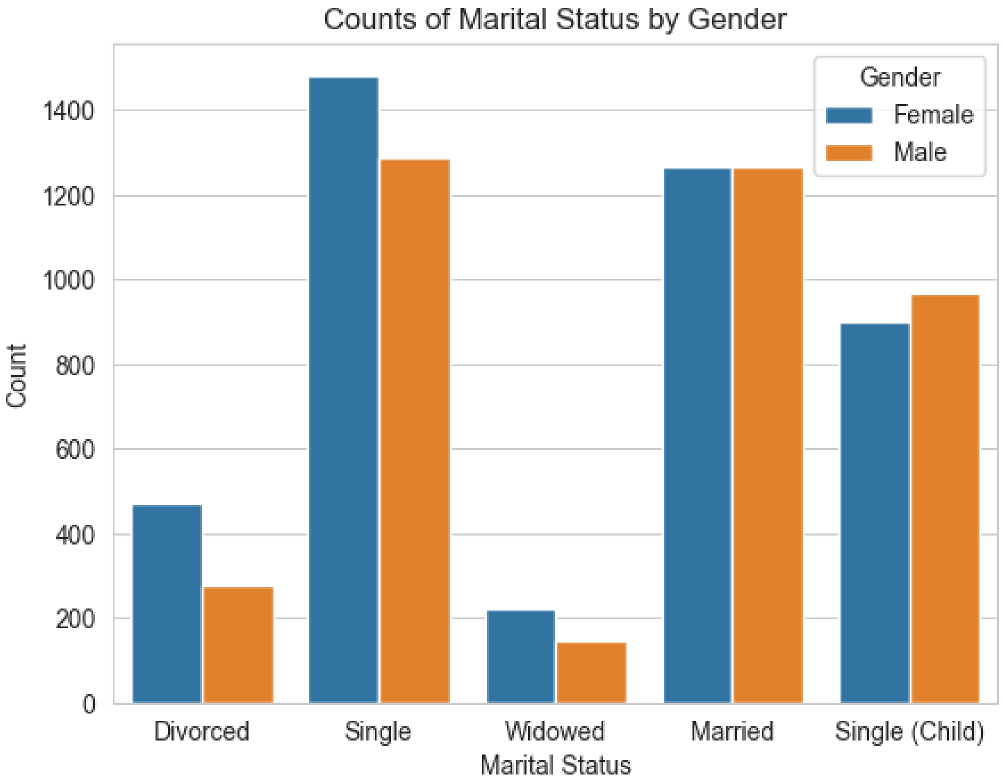
Visualising Occupation Feature



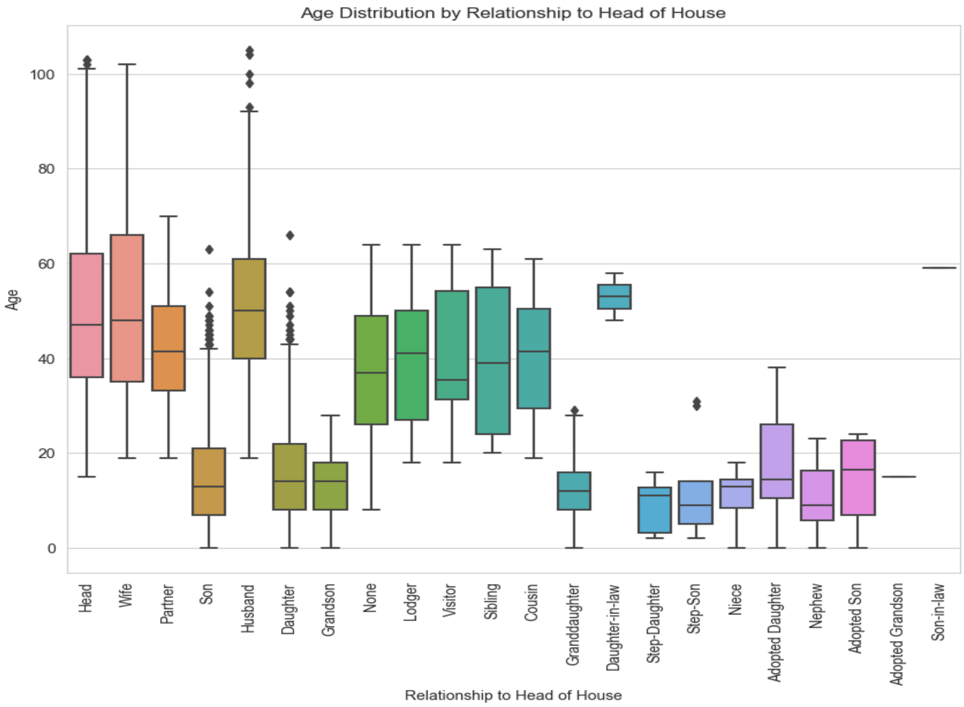
Visualising Religion Feature



Visualising Marital Status by Gender

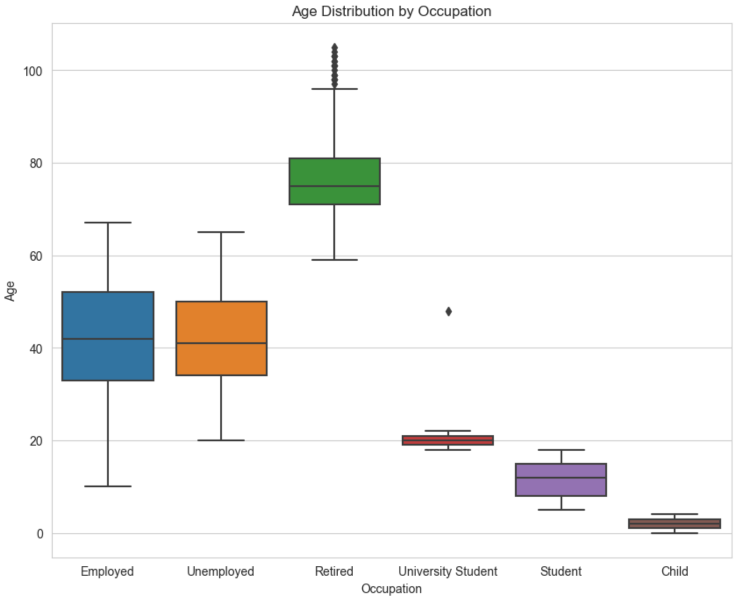


Visualising Age Distribution by Relationship to Head of House



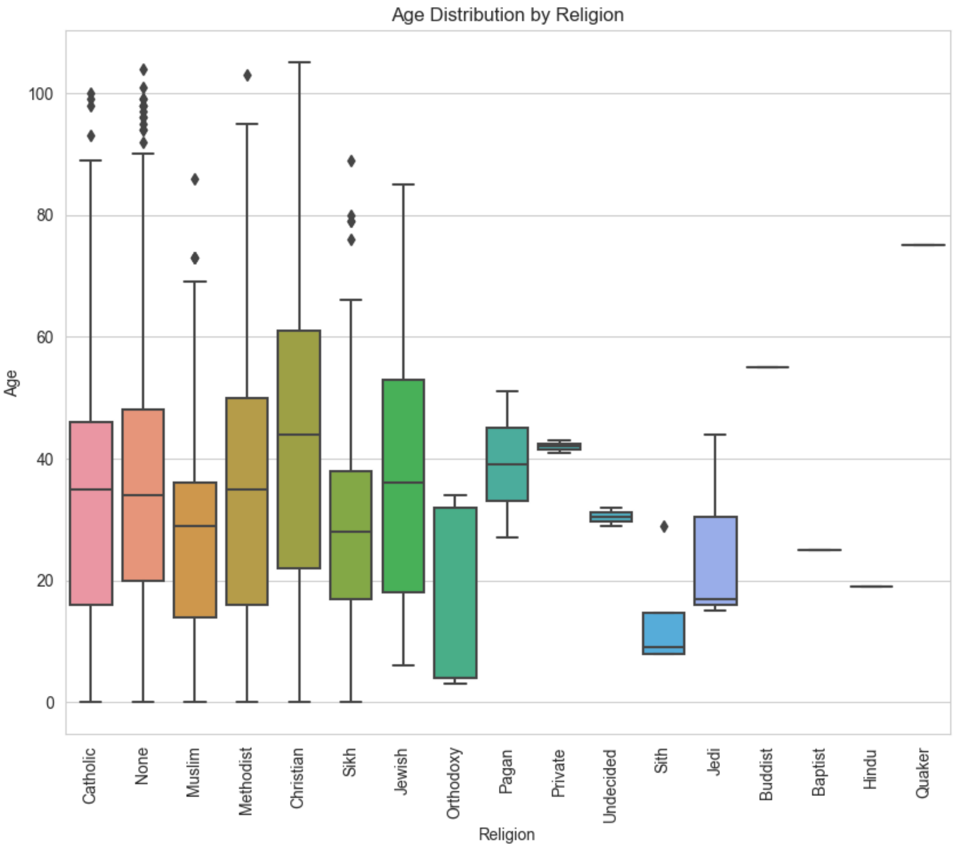
The boxplot above reveals some outliers in the Son and Daughter category of the Relationship to Head of House. Notably, some individuals above the age of 40 identified themselves as Sons and Daughters to their respective Heads of Houses, which is not a realistic scenario. However, the presence of outliers in the Head and Husband categories may not be inappropriate, as there could be a Head or Husband who is above 100 years old. However, it seems unlikely for a 30+ year-old Step-Son or a 30-year-old Granddaughter to be living with their grandparents.

Visualising Age Distribution by Occupation



Based on the boxplot above, we can observe outliers in the Retired category due to the presence of individuals who are above 100 years old. While this is not common, it is not necessarily inappropriate for someone in this category to live past 100 years. Similarly, it is not uncommon for university students to be around 50 years old, although this is not the norm. Additionally, there are individuals above 65 years of age who are employed, and this is not necessarily inappropriate as they may be engaged in contract jobs after their official retirement age.

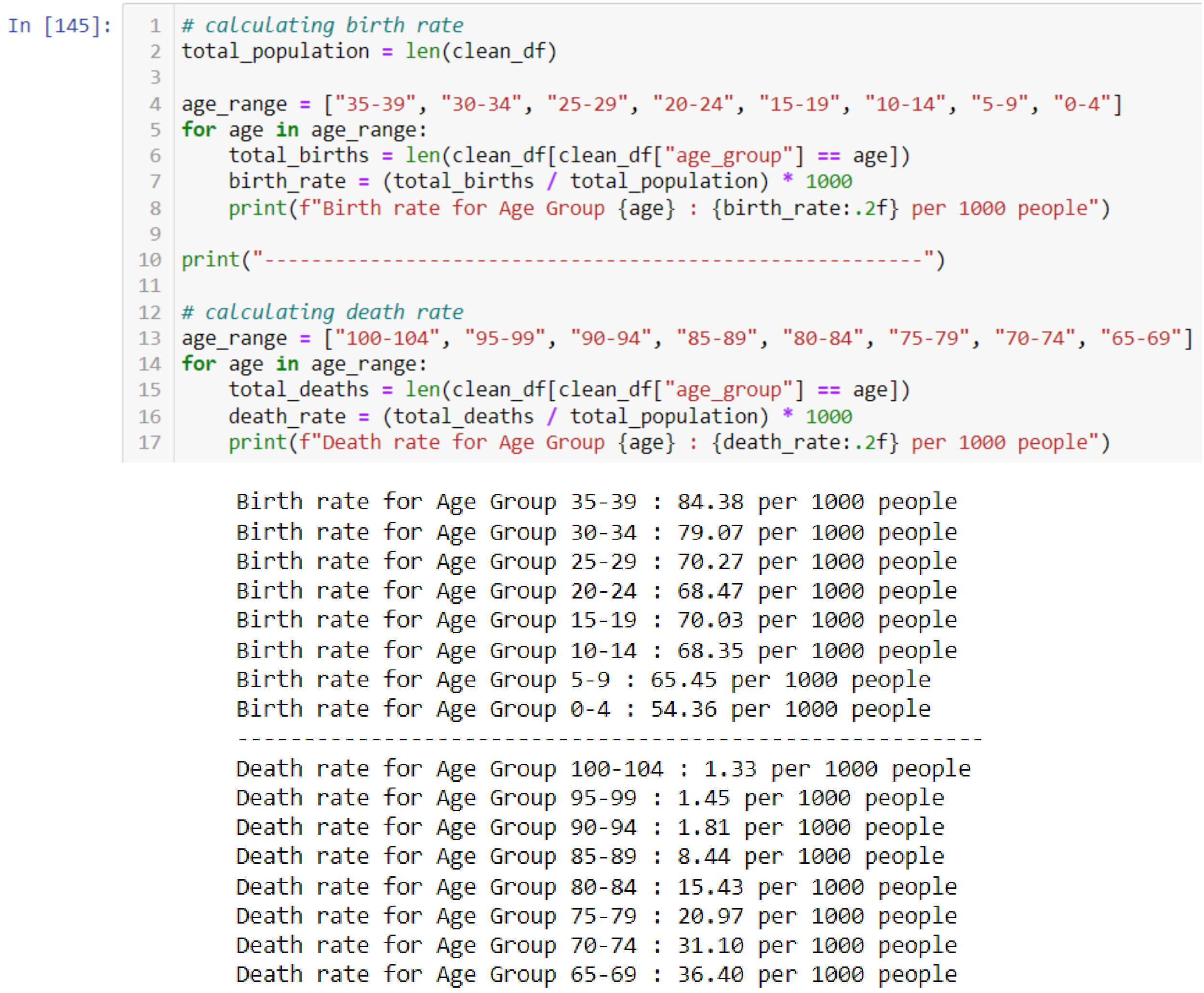
Visualising Age Distribution by Religion



The plot above reveals that the Catholic Religion is predominantly practiced by younger individuals, while the Sith Religion is mostly practiced by children. However, the fact that almost everyone who practices Sith Religion is a child seems unlikely. Overall, the plot suggests that religion is more common among young people in the population.

Data Analysis

Birth Rate & Death Rate



In [145]:

```python
# calculating birth rate
total_population = len(clean_df)

age_range = ["35-39", "30-34", "25-29", "20-24", "15-19", "10-14", "5-9", "0-4"]
for age in age_range:
    total_births = len(clean_df[clean_df["age_group"] == age])
    birth_rate = (total_births / total_population) * 1000
    print(f"Birth rate for Age Group {age} : {birth_rate:.2f} per 1000 people")

print("--------------------------------------------------------")

# calculating death rate
age_range = ["100-104", "95-99", "90-94", "85-89", "80-84", "75-79", "70-74", "65-69"]
for age in age_range:
    total_deaths = len(clean_df[clean_df["age_group"] == age])
    death_rate = (total_deaths / total_population) * 1000
    print(f"Death rate for Age Group {age} : {death_rate:.2f} per 1000 people")
```

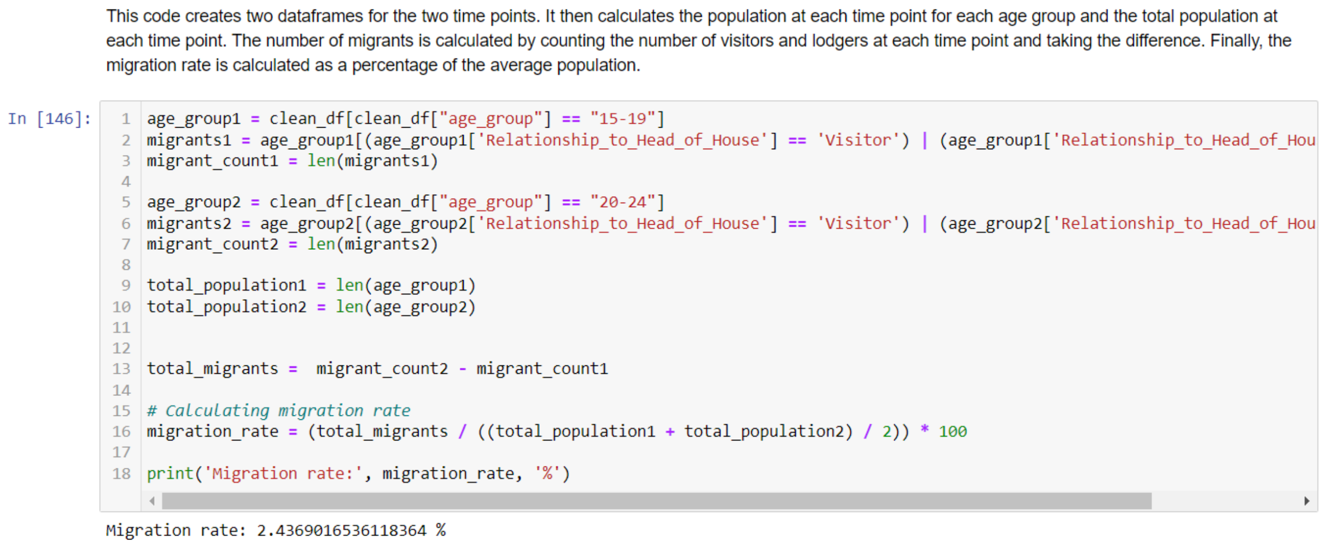
```
Birth rate for Age Group 35-39 : 84.38 per 1000 people
Birth rate for Age Group 30-34 : 79.07 per 1000 people
Birth rate for Age Group 25-29 : 70.27 per 1000 people
Birth rate for Age Group 20-24 : 68.47 per 1000 people
Birth rate for Age Group 15-19 : 70.03 per 1000 people
Birth rate for Age Group 10-14 : 68.35 per 1000 people
Birth rate for Age Group 5-9 : 65.45 per 1000 people
Birth rate for Age Group 0-4 : 54.36 per 1000 people
--------------------------------------------------------
Death rate for Age Group 100-104 : 1.33 per 1000 people
Death rate for Age Group 95-99 : 1.45 per 1000 people
Death rate for Age Group 90-94 : 1.81 per 1000 people
Death rate for Age Group 85-89 : 8.44 per 1000 people
Death rate for Age Group 80-84 : 15.43 per 1000 people
Death rate for Age Group 75-79 : 20.97 per 1000 people
Death rate for Age Group 70-74 : 31.10 per 1000 people
Death rate for Age Group 65-69 : 36.40 per 1000 people
```

Based on the estimated birth rate calculated, it is evident that there has been a decline in birth rate, leading to a decrease in population growth. Additionally, the estimated death rate indicates an increase in mortality.

When the birth rate is declining while the death rate is increasing, it generally indicates that the population is aging and that there are fewer births occurring. This demographic trend can have several implications, such as a smaller workforce to support an aging population, potential strain on social security and healthcare systems, and decreased economic growth. Additionally, this trend could lead to a decrease in overall population size if the death rate surpasses the birth rate, which could have further social and economic implications.

Migration Rate

This was calculated based on an assumption that the visitors and lodgers are migrants. The percentage of these migrants was calculated across 2 age groups that are known for migration. The calculated Migration rate: 2.44 %.



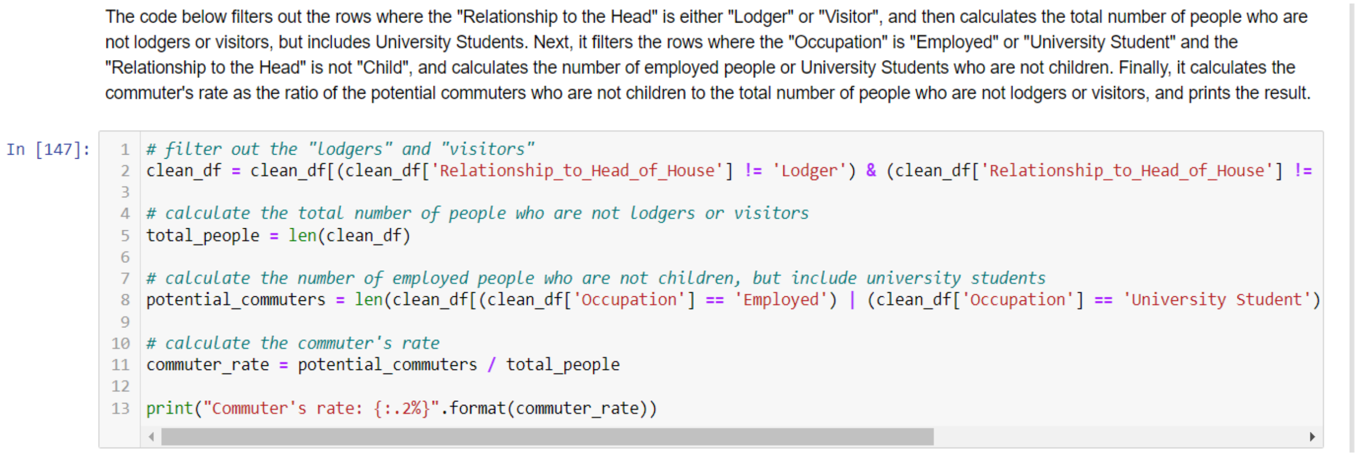
This code creates two dataframes for the two time points. It then calculates the population at each time point for each age group and the total population at each time point. The number of migrants is calculated by counting the number of visitors and lodgers at each time point and taking the difference. Finally, the migration rate is calculated as a percentage of the average population.

```
In [146]:
age_group1 = clean_df[clean_df["age_group"] == "15-19"]
migrants1 = age_group1[(age_group1['Relationship_to_Head_of_House'] == 'Visitor') | (age_group1['Relationship_to_Head_of_Hou
migrant_count1 = len(migrants1)

age_group2 = clean_df[clean_df["age_group"] == "20-24"]
migrants2 = age_group2[(age_group2['Relationship_to_Head_of_House'] == 'Visitor') | (age_group2['Relationship_to_Head_of_Hou
migrant_count2 = len(migrants2)

total_population1 = len(age_group1)
total_population2 = len(age_group2)


total_migrants =  migrant_count2 - migrant_count1

# Calculating migration rate
migration_rate = (total_migrants / ((total_population1 + total_population2) / 2)) * 100

print('Migration rate:', migration_rate, '%')
```

```
Migration rate: 2.4369016536118364 %
```

Commuter's Rate

The calculated commuter's rate is 59.2%. The calculation assumed that the University Students and the employed individuals are the major commuting population. The implication of the above result is that a significant proportion of the population travels to work or school from their homes, and this could have several implications for the community. For example, it could mean that there is a high demand for transportation services, such as buses or trains, during peak hours. It could also indicate that there is a need for more infrastructure or housing near employment centers to reduce commuting times and costs.

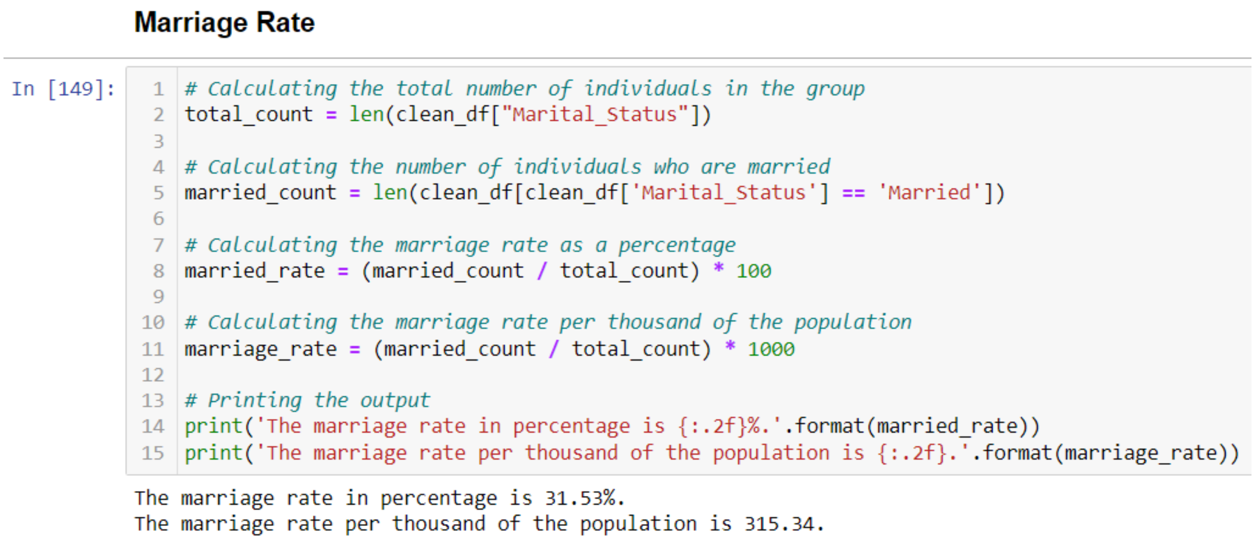


The code below filters out the rows where the "Relationship to the Head" is either "Lodger" or "Visitor", and then calculates the total number of people who are not lodgers or visitors, but includes University Students. Next, it filters the rows where the "Occupation" is "Employed" or "University Student" and the "Relationship to the Head" is not "Child", and calculates the number of employed people or University Students who are not children. Finally, it calculates the commuter's rate as the ratio of the potential commuters who are not children to the total number of people who are not lodgers or visitors, and prints the result.

```
In [147]:
# filter out the "lodgers" and "visitors"
clean_df = clean_df[(clean_df['Relationship_to_Head_of_House'] != 'Lodger') & (clean_df['Relationship_to_Head_of_House'] !=

# calculate the total number of people who are not lodgers or visitors
total_people = len(clean_df)

# calculate the number of employed people who are not children, but include university students
potential_commuters = len(clean_df[(clean_df['Occupation'] == 'Employed') | (clean_df['Occupation'] == 'University Student')

# calculate the commuter's rate
commuter_rate = potential_commuters / total_people

print("Commuter's rate: {:.2%}".format(commuter_rate))
```

Marriage & Divorce Rate

The divorce rate per thousand of the population is 83.93 while the marriage rate per thousand of the population is 315.34.

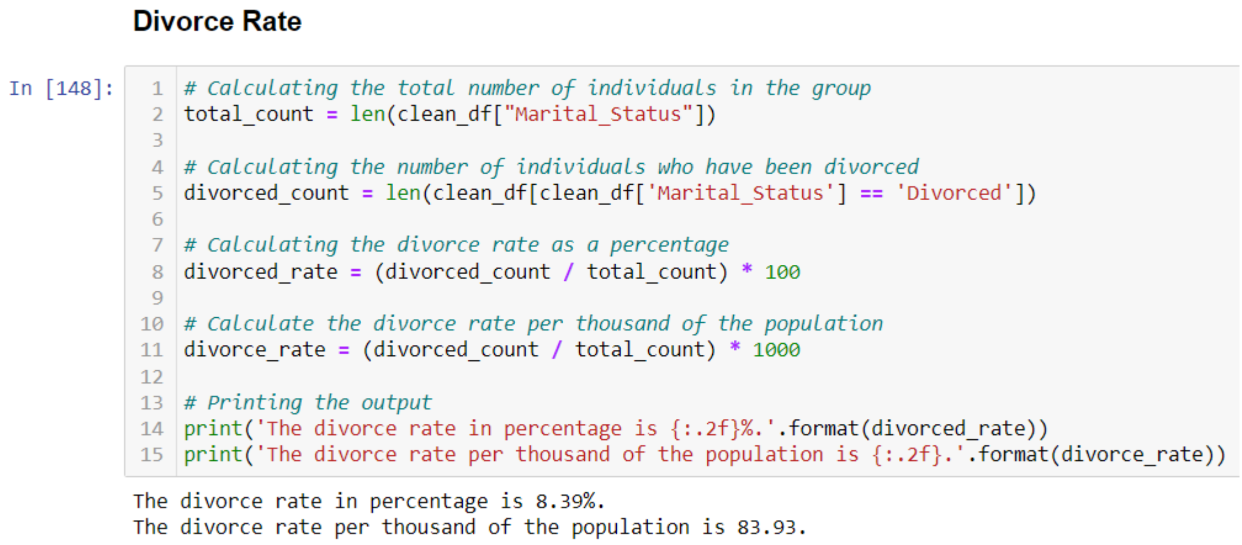


Marriage Rate

In [149]:

```python
# Calculating the total number of individuals in the group
total_count = len(clean_df["Marital_Status"])

# Calculating the number of individuals who are married
married_count = len(clean_df[clean_df['Marital_Status'] == 'Married'])

# Calculating the marriage rate as a percentage
married_rate = (married_count / total_count) * 100

# Calculating the marriage rate per thousand of the population
marriage_rate = (married_count / total_count) * 1000

# Printing the output
print('The marriage rate in percentage is {:.2f}%.'.format(married_rate))
print('The marriage rate per thousand of the population is {:.2f}.'.format(marriage_rate))
```

```
The marriage rate in percentage is 31.53%.
The marriage rate per thousand of the population is 315.34.
```



Divorce Rate

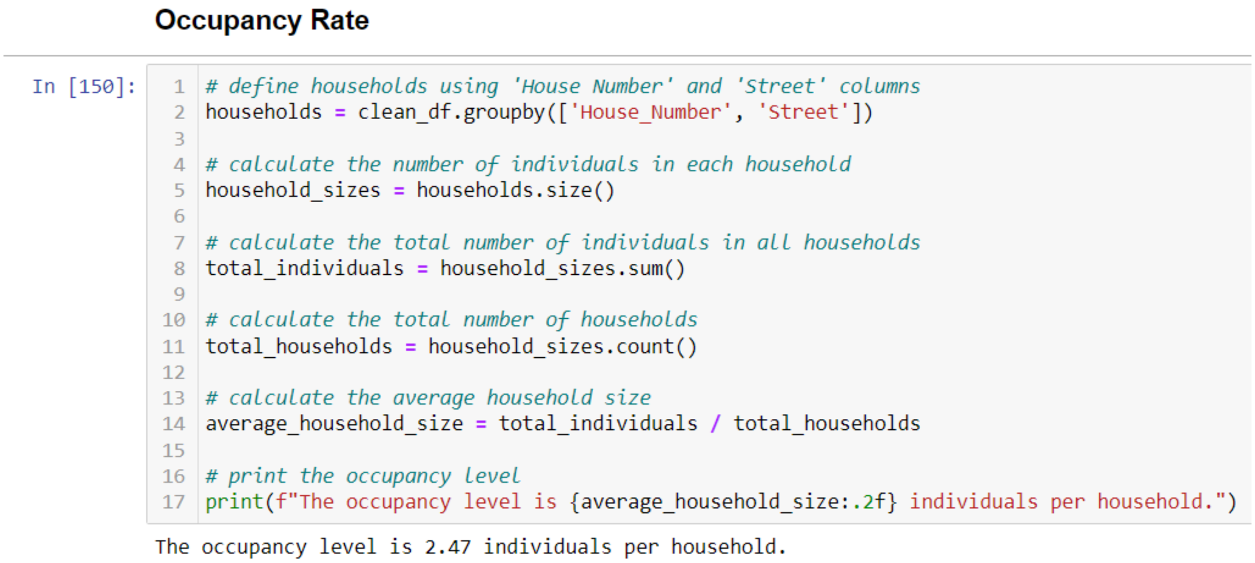
In [148]:

```python
# Calculating the total number of individuals in the group
total_count = len(clean_df["Marital_Status"])

# Calculating the number of individuals who have been divorced
divorced_count = len(clean_df[clean_df['Marital_Status'] == 'Divorced'])

# Calculating the divorce rate as a percentage
divorced_rate = (divorced_count / total_count) * 100

# Calculate the divorce rate per thousand of the population
divorce_rate = (divorced_count / total_count) * 1000

# Printing the output
print('The divorce rate in percentage is {:.2f}%.'.format(divorced_rate))
print('The divorce rate per thousand of the population is {:.2f}.'.format(divorce_rate))
```

```
The divorce rate in percentage is 8.39%.
The divorce rate per thousand of the population is 83.93.
```

Occupancy Rate

The calculated occupancy level is 2.47 individuals per household.



Occupancy Rate

In [150]:

```
# define households using 'House Number' and 'Street' columns
households = clean_df.groupby(['House_Number', 'Street'])

# calculate the number of individuals in each household
household_sizes = households.size()

# calculate the total number of individuals in all households
total_individuals = household_sizes.sum()

# calculate the total number of households
total_households = household_sizes.count()

# calculate the average household size
average_household_size = total_individuals / total_households

# print the occupancy level
print(f"The occupancy level is {average_household_size:.2f} individuals per household.")
```

```
The occupancy level is 2.47 individuals per household.
```

Recommendations

Based on my data analysis, I suggest the following recommendations for the unoccupied plot of land and potential investments:

(A)

1. Given the declining birth rate and the lack of demand for large family housing, high-density housing and low-density housing should not be prioritized. Additionally, emergency medical buildings may not be necessary due to the low number of injuries or expected pregnancies in the population. While religious buildings may have been a consideration, the town's top religion is Christian, with no specific denomination being a significant proportion of the population. Therefore, it may not be necessary to build a religious structure.

2. However, I strongly recommend building a train station on unoccupied land. This could bring numerous benefits, such as improved transportation options, a boost to the local economy, reduced traffic congestion, a better environment, and increased property values. This could be a valuable addition to a town that has never had a train station before.

(B)

1. With an unemployment rate of 6.1%, it is not significant enough to warrant investing in employment and training programs. Additionally, with a declining birth rate, it may not be wise to increase spending on schooling.

2. As there is no evidence that the town is expanding, it may not be necessary to invest in general infrastructure. The growth rate has been shrinking over time.

3. However, I strongly recommend that the town should invest in old age care. With employed individuals comprising 54.6% of the population, and the average age of the working class being about 45, this is a strong indication that the town will have increasing numbers of retired people in future years. Therefore, the town should allocate more funding for end-of-life care to cater to the needs of its aging population.

Bibliography

UK Child's Rights
Available online:
https://www.gov.uk/guidance/case-management-guidance/definitions#:~:text=We%20define%20a%20child%20as,legislation%20in%20England%20and%20Wales [Accessed 04/22/2023]

Working Retirement Age
Available online: https://www.gov.uk/working-retirement-pension-age [Accessed 04/20/2023]

1881 England Census
Available online: https://www.ancestry.co.uk/search/collections/7572/ [Accessed 04/16/2023]