# Exploring categorical variables

## Contingency tables and bar plots

# Import data

In [1]:

```
import pandas as pd

ROOT = "https://raw.githubusercontent.com/kirenz/modern-statistics/main/data/"
DATA = "loans.csv"

df = pd.read_csv(ROOT + DATA)
```

In [2]:

```
# Show head of dataframe for our two variables
df[["homeownership", "application_type"]].head()
```

Out[2]:

| | homeownership | application_type |
|---|---|---|
| 0 | mortgage | individual |
| 1 | rent | individual |
| 2 | rent | individual |
| 3 | rent | individual |
| 4 | rent | joint |

In [3]:

```
# Show tail of dataframe for our two variables
df[["homeownership", "application_type"]].tail()
```

Out[3]:

| | homeownership | application_type |
|---|---|---|
| 9995 | rent | individual |
| 9996 | mortgage | individual |
| 9997 | mortgage | joint |
| 9998 | mortgage | individual |
| 9999 | rent | individual |

In [4]:

```
# Show info about variables
df[["homeownership", "application_type"]].info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 2 columns):
 #   Column            Non-Null Count  Dtype 
---  ------            --------------  ----- 
 0   homeownership     10000 non-null  object
 1   application_type  10000 non-null  object
dtypes: object(2)
memory usage: 156.4+ KB
```

In [5]:

```
# Change data format from object to category
df['homeownership'] = df['homeownership'].astype("category")
df.application_type = df.application_type.astype("category")

# Show info
df[["homeownership", "application_type"]].info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 2 columns):
 #   Column            Non-Null Count  Dtype   
---  ------            --------------  -----   
 0   homeownership     10000 non-null  catego
ry
 1   application_type  10000 non-null  catego
ry
dtypes: category(2)
memory usage: 19.9 KB
```

In [6]:

```
# Show levels of categorical variables
print("homeownership:", df['homeownership'].cat.categories, "\n")
print("application_type:", df['application_type'].cat.categories)
```

```
homeownership: Index(['mortgage', 'own', 'ren
t'], dtype='object')

application_type: Index(['individual', 'join
t'], dtype='object')
```

In [7]:

```
# Summarizing the frequencies for each value
print(df["homeownership"].value_counts(), "\n")
print(df["application_type"].value_counts())
```

```
mortgage    4789
rent        3858
own         1353
Name: homeownership, dtype: int64

individual    8505
joint         1495
Name: application_type, dtype: int64
```

# Contingency table

In [8]:

```
# contingency table for application type and homeownership.
pd.crosstab(df.application_type , df.homeownership,  margins=False)
```

Out[8]:

| homeownership | mortgage | own | rent |
|---|---|---|---|
| application_type | | | |
| individual | 3839 | 1170 | 3496 |
| joint | 950 | 183 | 362 |

In [9]:

```
# contingency table for application type and homeownership.
pd.crosstab(df.application_type , df.homeownership,  margins=True)
```

Out[9]:

| homeownership | mortgage | own | rent | All |
|---|---|---|---|---|
| application_type | | | | |
| individual | 3839 | 1170 | 3496 | 8505 |
| joint | 950 | 183 | 362 | 1495 |
| All | 4789 | 1353 | 3858 | 10000 |

# Bar plot

In [10]:

```
%matplotlib inline

import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

sns.set_theme(style="ticks", color_codes=True)

# Colors
blue = "#3F83F4"
blue_dark = "#062089"
blue_light = "#8DC0F6"
blue_lighter = "#BBE4FA"
grey = "#9C9C9C"
grey_dark = "#777777"
grey_light = "#B2B2B2"
orange = "#EF8733"

colors_blue = [blue_dark, blue, blue_light]
```

In [11]:

```
# Counts of values of the homeownership variable.
sns.catplot(x="homeownership",
            kind = "count",
            palette=colors_blue,
            data=df)

plt.title("Bar chart")
plt.xlabel("Homeownership")

plt.show();
```