```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

## context:

Austo Motor Company is a leading car manufacturer specializing in SUV, Sedan, and Hatchback models. In its recent board meeting, concerns were raised by the members on the efficiency of the marketing campaign currently being used. The board decides to rope in an analytics professional to improve the existing campaign.

## Objective:

They want to analyze the data to get a fair idea about the demand of customers which will help them in enhancing their customer experience. Suppose you are a Data Scientist at the company and the Data Science team has shared some of the key questions that need to be answered. Perform the data analysis to find answers to these questions that will help the company to improve the business.

## Key questions

*Do men tend to prefer SUVs more compared to women? *What is the likelihood of a salaried person buying a Sedan? *What evidence supports Sheldon Cooper's claim that a salaried male is an easier target for an SUV sale over a Sedan sale? *How does the amount spent on purchasing *automobiles vary by gender? *How much money was spent on purchasing automobiles by individuals who took a personal loan? *How does having a working partner influence the purchase of higher-priced cars

## Data Description

- Age: The age of the individual in years.
- Gender: The gender of the individual, categorized as male or female.
- Profession: The occupation or profession of the individual.
- Marital_status: The marital status of the individual, such as married &, single
- Education: The educational qualification of the individual Graduate and Post Graduate
- No_of_Dependents: The number of dependents (e.g., children, elderly parents) that the individual supports financially.
- Personal_loan: A binary variable indicating whether the individual has taken a personal loan "Yes" or "No"
- House_loan: A binary variable indicating whether the individual has taken a housing loan "Yes" or "No"
- Partner_working: A binary variable indicating whether the individual's partner is employed "Yes" or "No"
- Salary: The individual's salary or income.
- Partner_salary: The salary or income of the individual's partner, if applicable.
- Total_salary: The total combined salary of the individual and their partner (if applicable).
- Price: The price of a product or service.
- Make: The type of automobile

## Importing the necessay libraries

```
import pandas as pd
import numpy as np

#libraries help in visuaization
import matplotlib.pyplot as plt
import seaborn as sns
```

Double-click (or enter) to edit

## Loading the dataset

```
data = pd.read_csv('/content/drive/MyDrive/projects/coded project python/austo_automobile+%282%29+%281%29.csv')
```

```
#copying data to another variable to avoid any changes to original data
df = data.copy()
```

## Data overview

```
#getting the first five rows of data
df.head()
```

| | Age | Gender | Profession | Marital_status | Education | No_of_Dependents | Personal_loa |
|---|---|---|---|---|---|---|---|
| **0** | 53 | Male | Business | Married | Post Graduate | 4 | N |
| **1** | 53 | Femal | Salaried | Married | Post Graduate | 4 | Ye |
| **2** | 53 | Female | Salaried | Married | Post Graduate | 3 | N |

Next steps: View recommended plots

```
df.tail()
```

| | Age | Gender | Profession | Marital_status | Education | No_of_Dependents | Personal_ |
|---|---|---|---|---|---|---|---|
| **1576** | 22 | Male | Salaried | Single | Graduate | 2 | |
| **1577** | 22 | Male | Business | Married | Graduate | 4 | |
| **1578** | 22 | Male | Business | Single | Graduate | 2 | |
| **1579** | 22 | Male | Business | Married | Graduate | 3 | |
| **1580** | 22 | Male | Salaried | Married | Graduate | 4 | |

```
#checking the shape of data
df.shape
```

```
(1581, 14)
```

- This dataset has 1581 rows and 14 columns

```
#checking the data type of columns for datasets
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1581 entries, 0 to 1580
Data columns (total 14 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   Age               1581 non-null   int64  
 1   Gender            1528 non-null   object 
 2   Profession        1581 non-null   object 
 3   Marital_status    1581 non-null   object 
 4   Education         1581 non-null   object 
 5   No_of_Dependents  1581 non-null   int64  
 6   Personal_loan     1581 non-null   object 
 7   House_loan        1581 non-null   object 
 8   Partner_working   1581 non-null   object 
 9   Salary            1581 non-null   int64  
 10  Partner_salary    1475 non-null   float64
 11  Total_salary      1581 non-null   int64  
 12  Price             1581 non-null   int64  
 13  Make              1581 non-null   object 
dtypes: float64(1), int64(5), object(8)
memory usage: 173.0+ KB
```

- all the columns have 1581 observation except Gender and partner_salary
- Gender has 1528 observation
- The partners_salary has 1475 observation
- The object type contains category
- 8 columns are object and 6 are numerical columns

## Statistical summary

```
df.isnull().sum()
```

```
Age                  0
Gender              53
Profession           0
Marital_status       0
Education            0
No_of_Dependents     0
Personal_loan        0
House_loan           0
Partner_working      0
Salary               0
Partner_salary     106
Total_salary         0
Price                0
Make                 0
dtype: int64
```

- there are 53 missing values in gender
- 106 values are misisng in partner_salary

```
#making the missing and nan values in python
df = pd.read_csv('/content/drive/MyDrive/projects/coded project python/austo_automobile+%282%29+%281%29.csv',na_values=['missing','inf'
```

```
#checking the duplicate values
df.duplicated().sum()
```

```
0
```

- There are no duplicate values in the dataset

```
df.Gender.value_counts()
```

```
Gender
Male      1199
Female     327
Femal        1
Femle        1
Name: count, dtype: int64
```

```
#Recorrecting the misspelled female
df['Gender'] = df['Gender'].replace(['Femal','Femle'],'Female')
```

```
df.Gender.value_counts()
```

```
Gender
Male      1199
Female     329
Name: count, dtype: int64
```

```
df['Gender'].isna().sum()
```

```
53
```

## Missing value treatment

- treating missing value for gender, since gender is a categorical data we can impute it using mode

```
mode_g = df['Gender'].mode()[0]
mode_g
```

```
'Male'
```

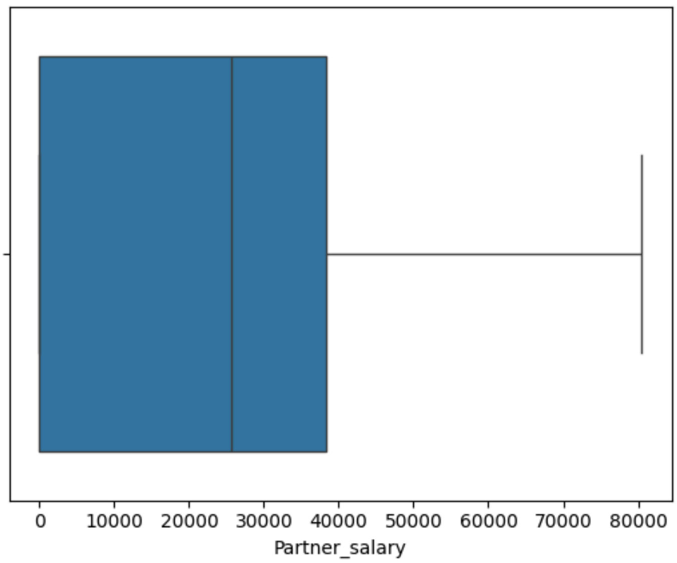
- The mode of the gender is male , hence now replacing the null values in gender column by 'male'

```
df['Gender'].fillna(mode_g,inplace=True)
```

```
df.loc[data['Partner_salary'].isnull()==True]
```

| | Age | Gender | Profession | Marital_status | Education | No_of_Dependents | Personal_ |
|---|---|---|---|---|---|---|---|
| **40** | 53 | Female | Salaried | Married | Graduate | 1 | |
| **43** | 52 | Male | Salaried | Married | Post Graduate | 3 | |
| **49** | 52 | Female | Business | Married | Post Graduate | 4 | |
| **59** | 54 | Male | Salaried | Married | Graduate | 3 | |
| **111** | 48 | Female | Business | Married | Graduate | 3 | |
| **...** | ... | ... | ... | ... | ... | ... | |
| **1559** | 22 | Male | Business | Married | Post Graduate | 3 | |
| **1567** | 22 | Male | Salaried | Single | Graduate | 0 | |
| **1568** | 22 | Male | Salaried | Married | Graduate | 3 | |
| **1577** | 22 | Male | Business | Married | Graduate | 4 | |

```
#checking the distribution before deciding weather to use mean,median or mode imputation
sns.boxplot(data=df,x='Partner_salary');
```



```
df['Partner_salary'].skew()
```

```
0.33825489824593036
```

- since the data is skewed median should be used
- the data is right skewed

```
#Given that salary-related fields often have outliers, using the median might be more robust than the mean
#Calculate the Median of the "Partner_salary" Column
median_partner_sal= df['Partner_salary'].median()
median_partner_sal
```

```
25600.0
```

- Filling missing values with median

```
df['Partner_salary'].fillna(median_partner_sal,inplace=True)
```

```
df.isnull().sum()
```

```
Age                 0
Gender              0
Profession          0
Marital_status      0
Education           0
No_of_Dependents    0
Personal_loan       0
```

```
House_loan         0
Partner_working    0
Salary             0
Partner_salary     0
Total_salary       0
Price              0
Make               0
dtype: int64
```

we have treated the missing values

## Statstical summary

```
df.describe().T
```

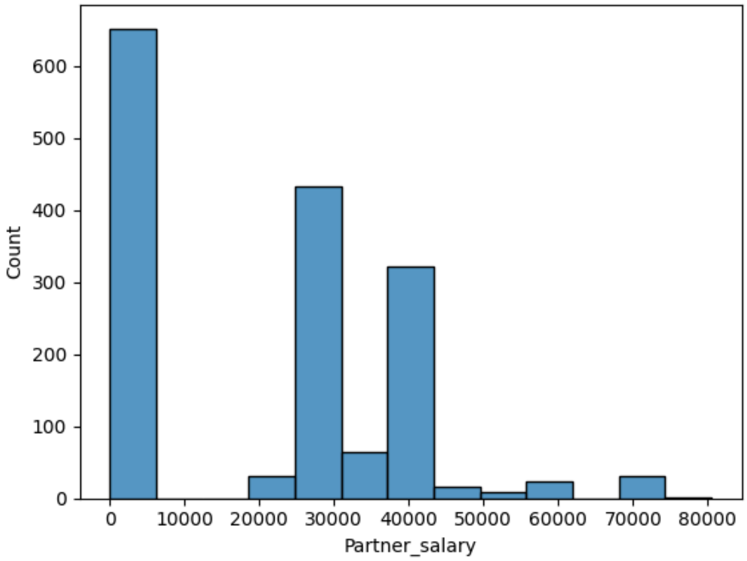
| | count | mean | std | min | 25% | 50% | 75 |
|---|---|---|---|---|---|---|---|
| **Age** | 1581.0 | 31.922201 | 8.425978 | 22.0 | 25.0 | 29.0 | 38. |
| **No_of_Dependents** | 1581.0 | 2.457938 | 0.943483 | 0.0 | 2.0 | 2.0 | 3. |
| **Salary** | 1581.0 | 60392.220114 | 14674.825044 | 30000.0 | 51900.0 | 59500.0 | 71800. |
| **Partner_salary** | 1581.0 | 20585.895003 | 18952.938643 | 0.0 | 0.0 | 25600.0 | 38000. |
| **Total_salary** | 1581.0 | 79625.996205 | 25545.857768 | 30000.0 | 60500.0 | 78000.0 | 95900. |
| **Price** | 1581.0 | 35597.722960 | 13633.636545 | 18000.0 | 25000.0 | 31000.0 | 47000. |

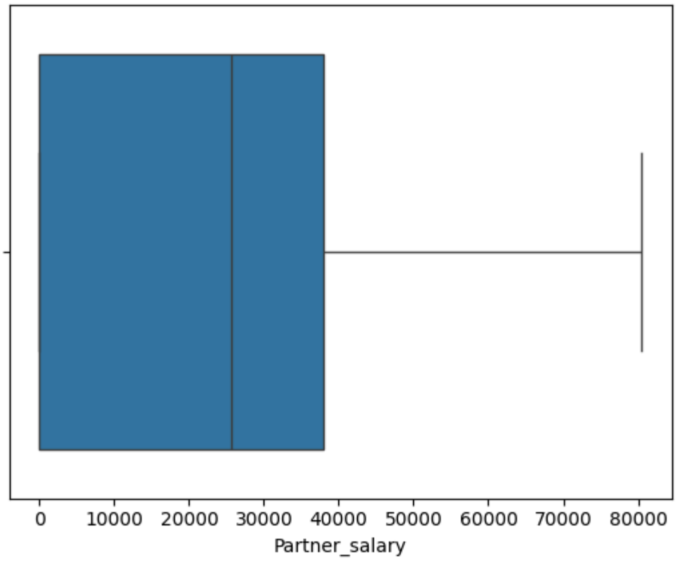
## Univariate analysis

- Categorical data = Gender, Profession, Marital_status, Education, Personal_loan, House_loan, Partner_working, Make
- Numerical data = Age, No_of_Dependents, Salary, Partner_salary, Total_salary, Price

- lets check the distribution for numerical column

```
sns.histplot(data=df,x='Partner_salary')
plt.show()
sns.boxplot(data=df,x='Partner_salary')
plt.show()
```





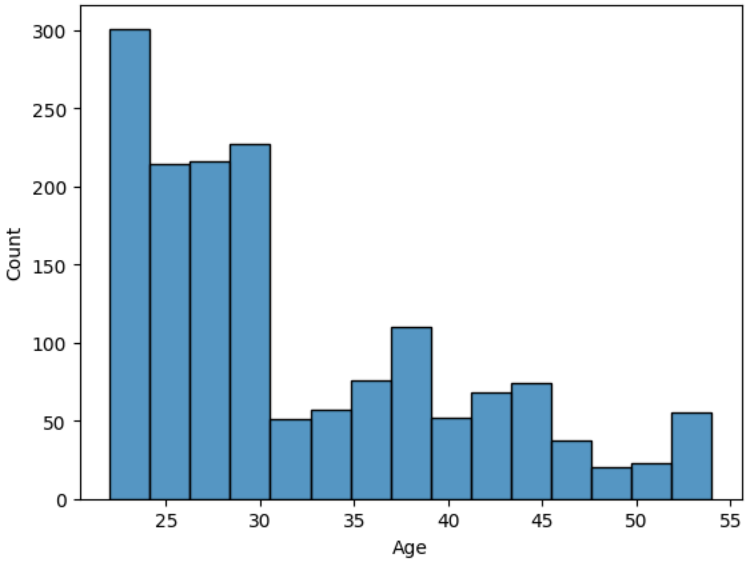
```
df['Partner_salary'].skew()
```

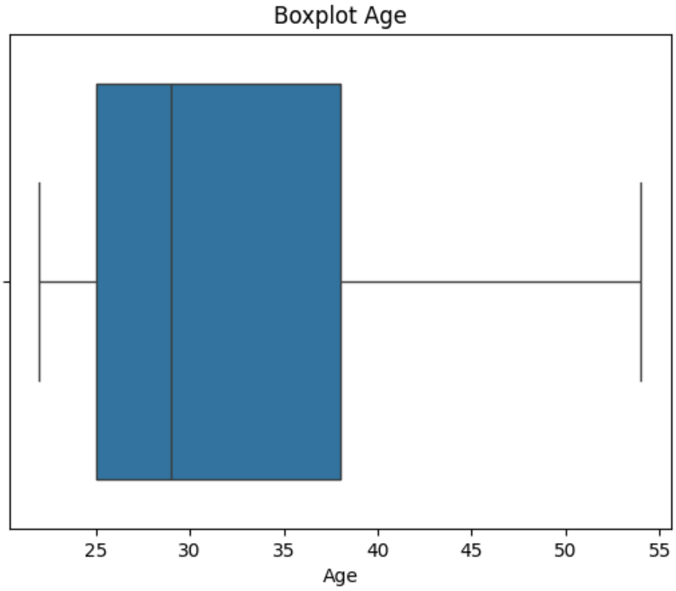
```
0.2919502622828292
```

- This distribution is right skewed
- the avergae salary of partner is around 250000

**Observation on age**

```
sns.histplot(data=df,x='Age')
plt.show()
sns.boxplot(data=df,x='Age')
plt.title('Boxplot Age')
plt.show()
```

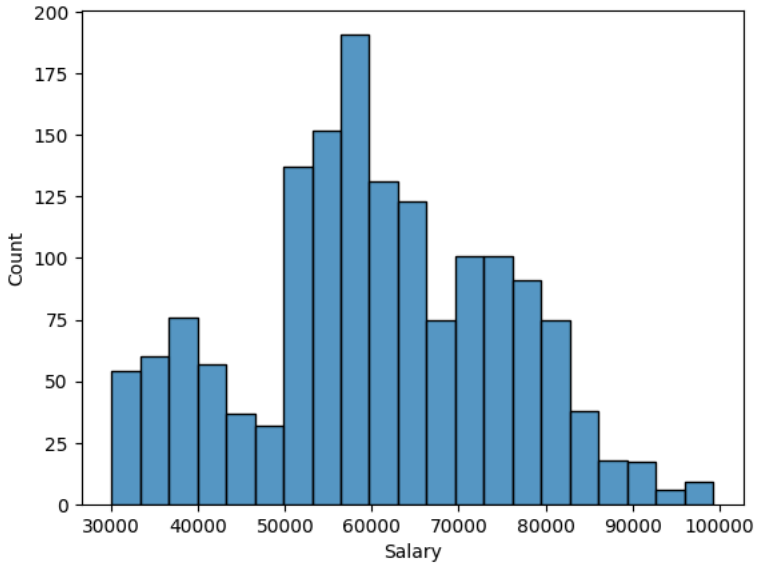


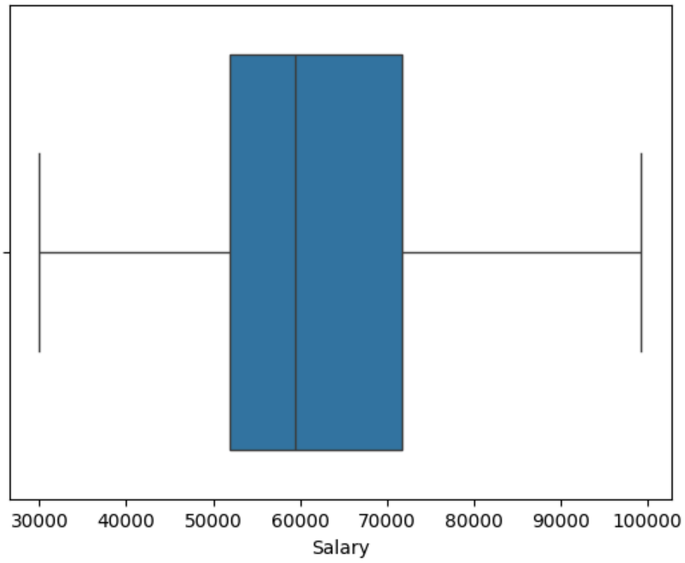


- there are no outliers as such
- but the min age is 22 and max is around 55
- the avergae age is 30
- the distribution is right skewed

### Observation on salary

```
sns.histplot(data=df,x='Salary')
plt.show()
sns.boxplot(data=df,x='Salary')
plt.show()
```

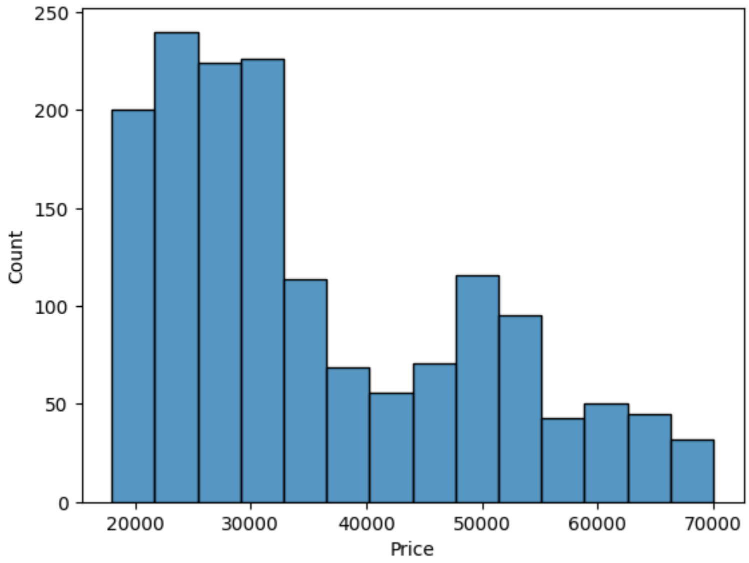


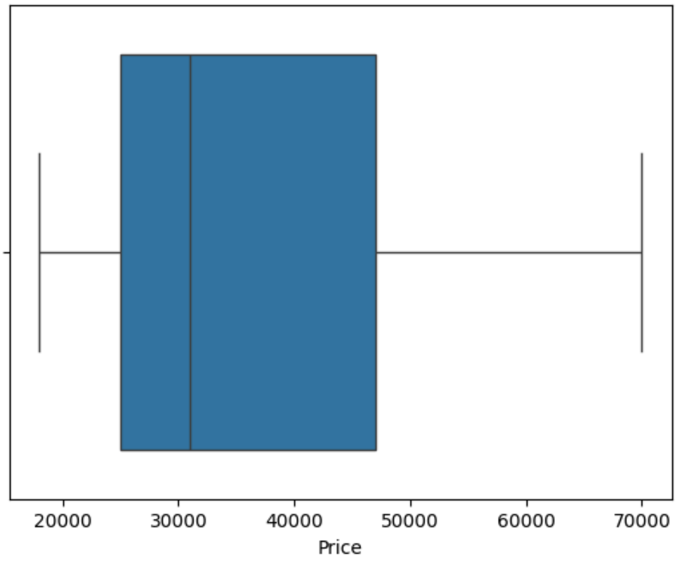


- min salary is 30k and max is 95k
- the distrubution is equally skewed
- avergae salary is 60k

**observation on price**

```
sns.histplot(data=df,x='Price')
plt.show()
sns.boxplot(data=df,x='Price')
plt.show()
```



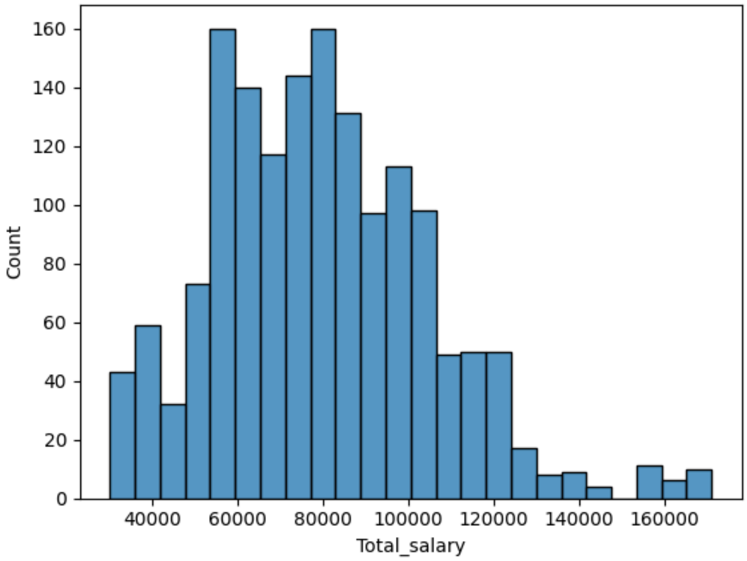


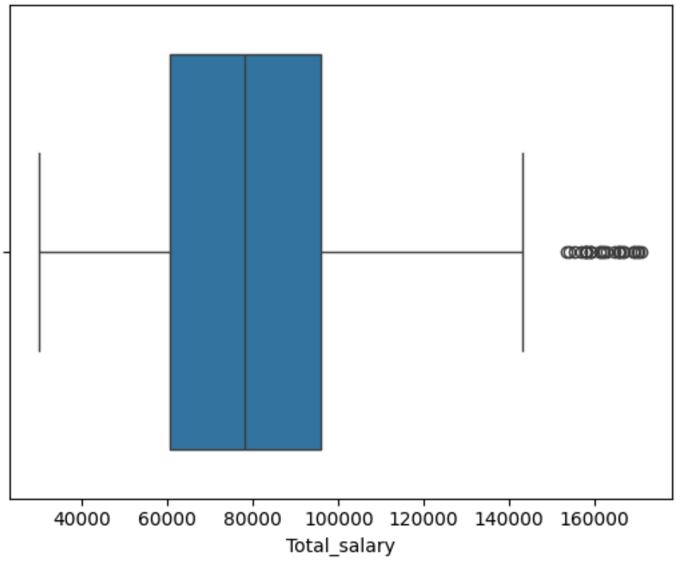
- There is no potential outlier, he distribution is right skewed
- the averge price is 30k

Double-click (or enter) to edit

**observation on total salary**

```
sns.histplot(data=df,x='Total_salary')
plt.show()
sns.boxplot(data=df,x='Total_salary')
plt.show()
```



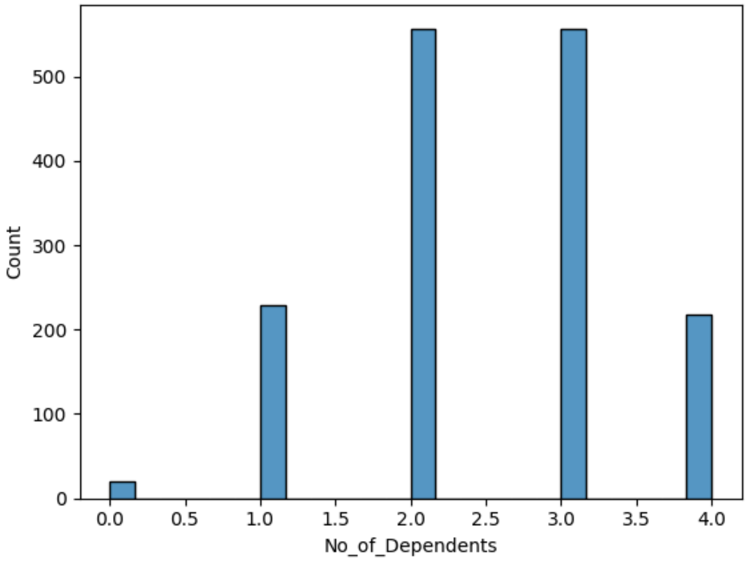


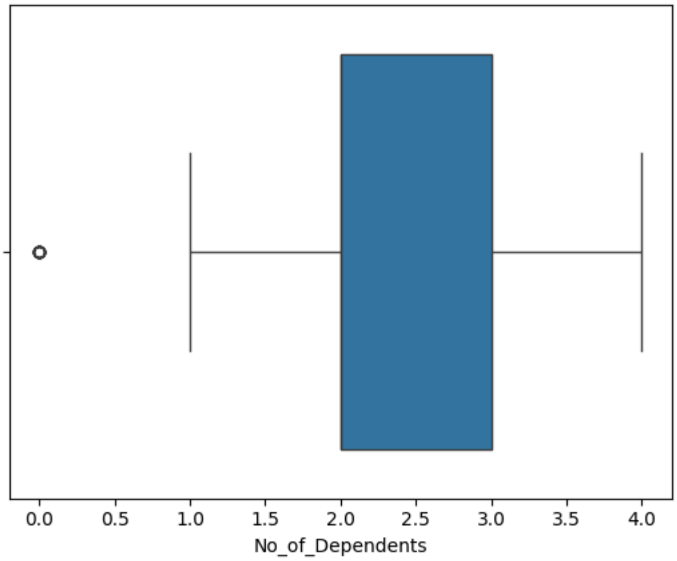
- there are many outlier in total salary data
- this distribution is right skewed
- the average of total salary is 80k

Double-click (or enter) to edit

**observation on no of dependencies**

```
sns.histplot(data=df,x='No_of_Dependents')
plt.show()
sns.boxplot(data=df,x='No_of_Dependents')
plt.show()
```

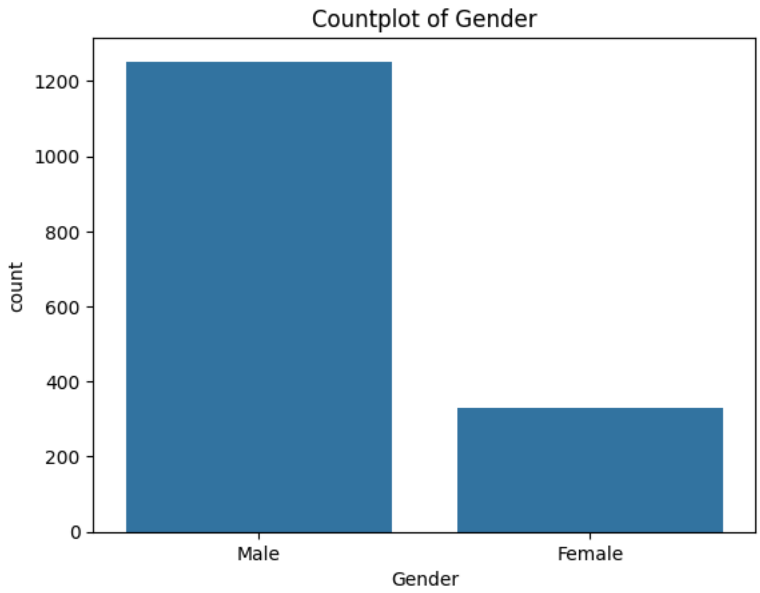


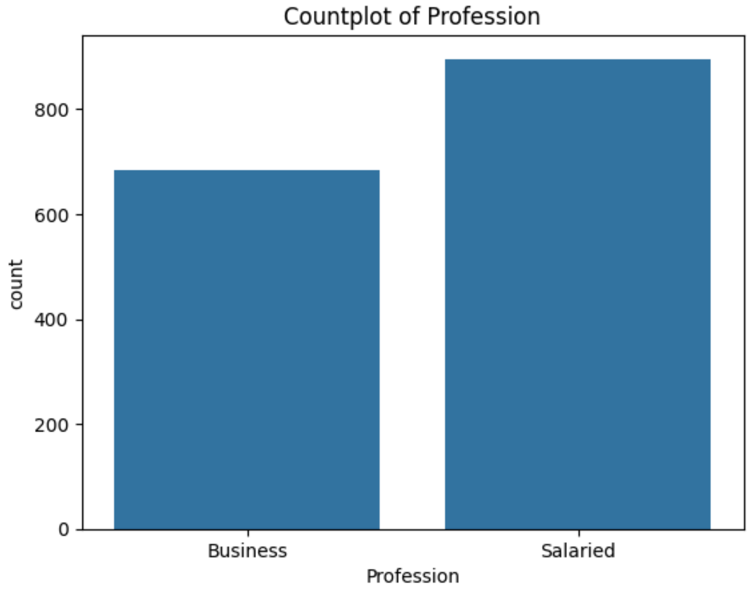


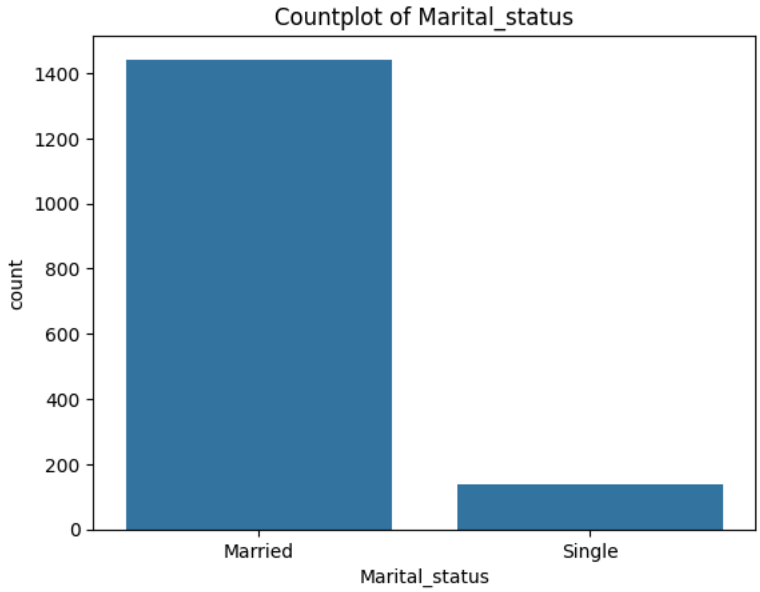
- this data has 1 outlier
- but the data is equally distributed
- max number of depndencies is 4
- the min number of dependencies is 1

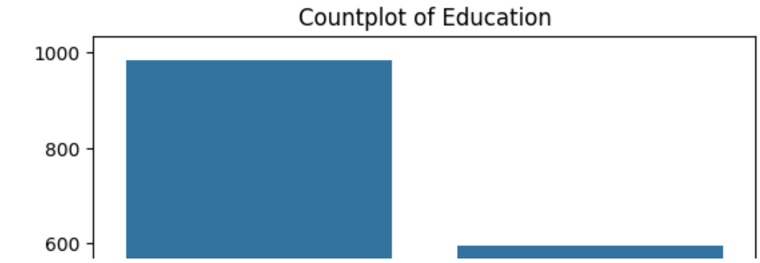
**Create countplots for each categorical column**

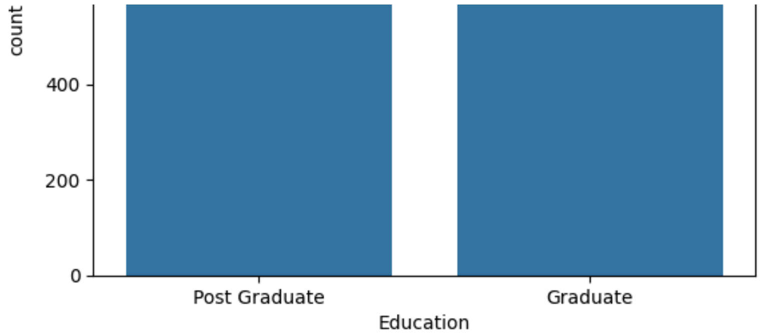
```
cat_columns = ['Gender', 'Profession', 'Marital_status', 'Education', 'Personal_loan', 'House_loan']

for col in cat_columns:
    sns.countplot(data=df, x=col)
    plt.title(f'Countplot of {col}')
    plt.show()
```

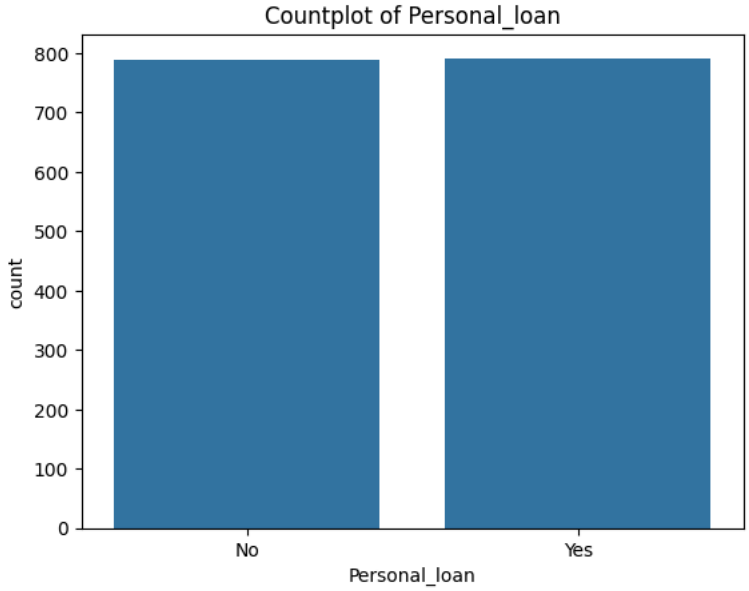

Countplot of Gender

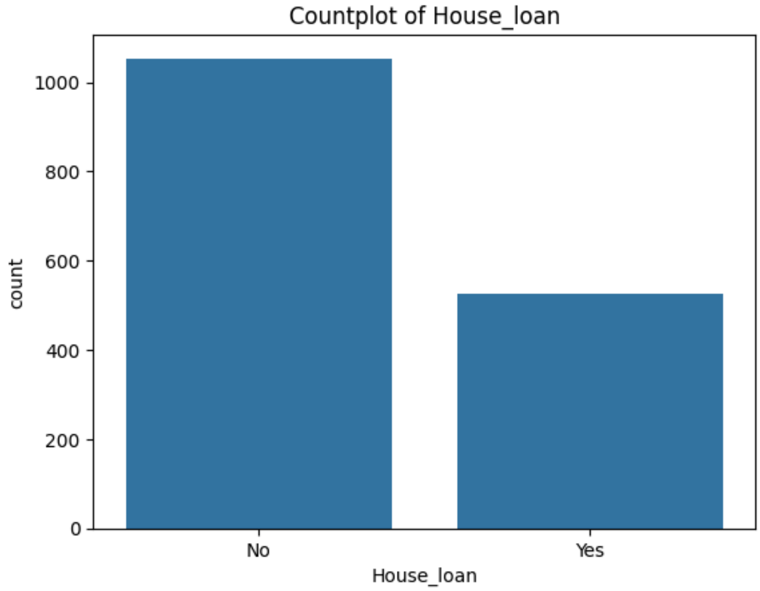

Countplot of Profession


Countplot of Marital_status


Countplot of Education





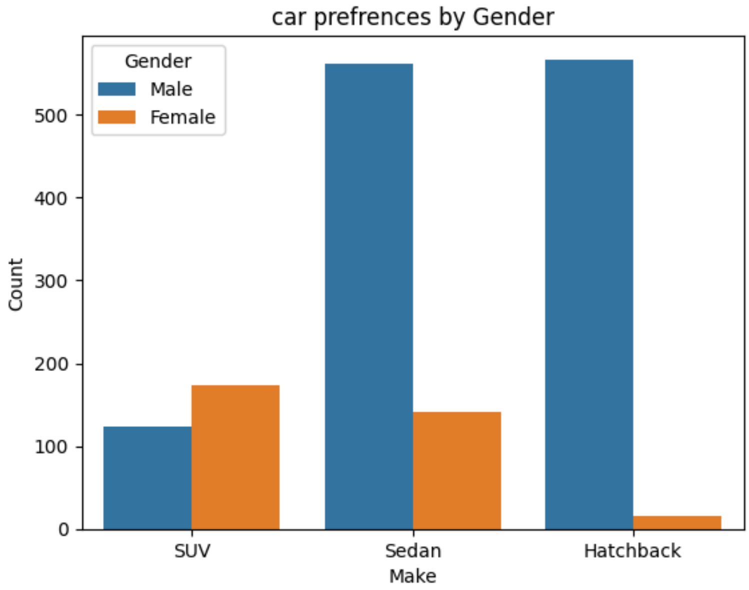


- There are more males then female
- most people are doing jobs than business by profession
- many people are married and very few are single
- more people have done postgraduation than graduation
- there are equal number of people taking personal loan and not taking loan
- there are very few people who took house loan

## Bivariate analysis

**Gender vs make**

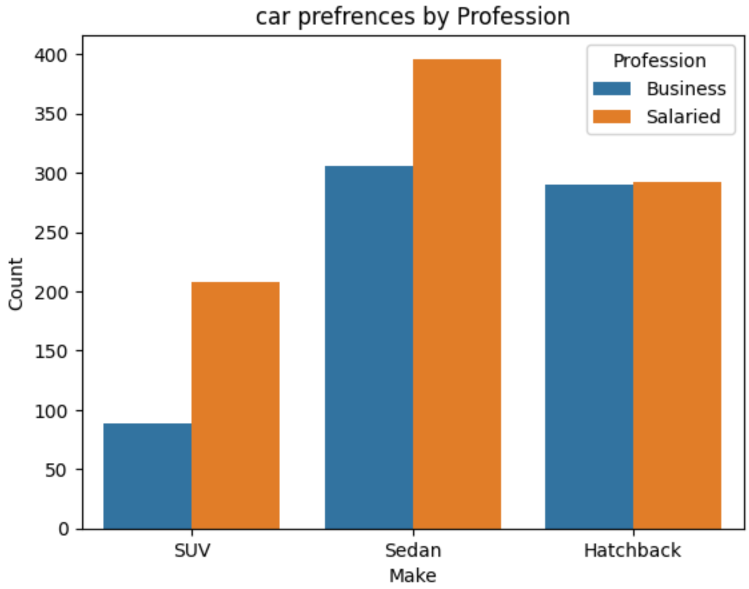
```
sns.countplot(data=df,x='Make',hue='Gender')
plt.title('car prefrences by Gender')
plt.xlabel('Make')
plt.ylabel('Count')
plt.show()
```



- most female(190) prefer suv compared to male(120)
- sedan and hatchback is prefered by the male by than female

**profession vs make**

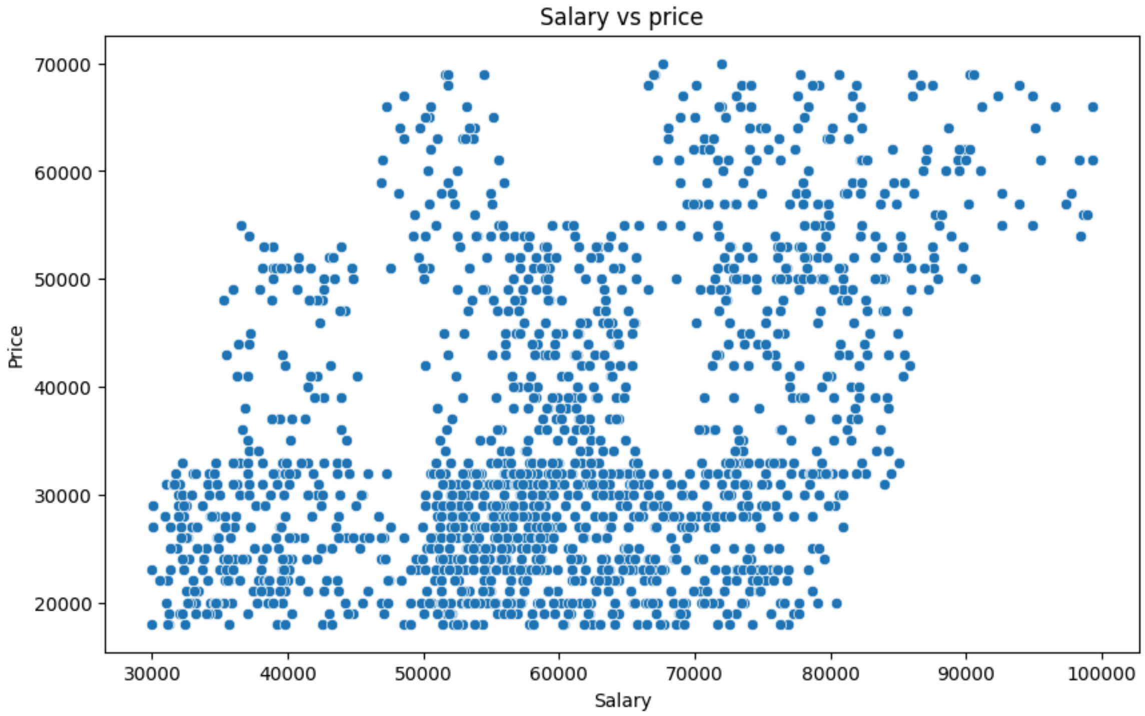
```
sns.countplot(data=df,x='Make',hue='Profession')
plt.title('car prefrences by Profession')
plt.xlabel('Make')
plt.ylabel('Count')
plt.show()
```



- Suvs are prefered by people who are into jobs then business
- Sedan is prefered by people who are into jobs than business
- Hatchback are prefered almost equally by both those unto jobs and business

**salary vs purchase price** To analyze spending patterns and how much different salary groups are spending.

```
#using scatter plot
plt.figure(figsize=(10,6))
sns.scatterplot(data=df,x='Salary',y='Price')
plt.title('Salary vs price')
plt.show()
```

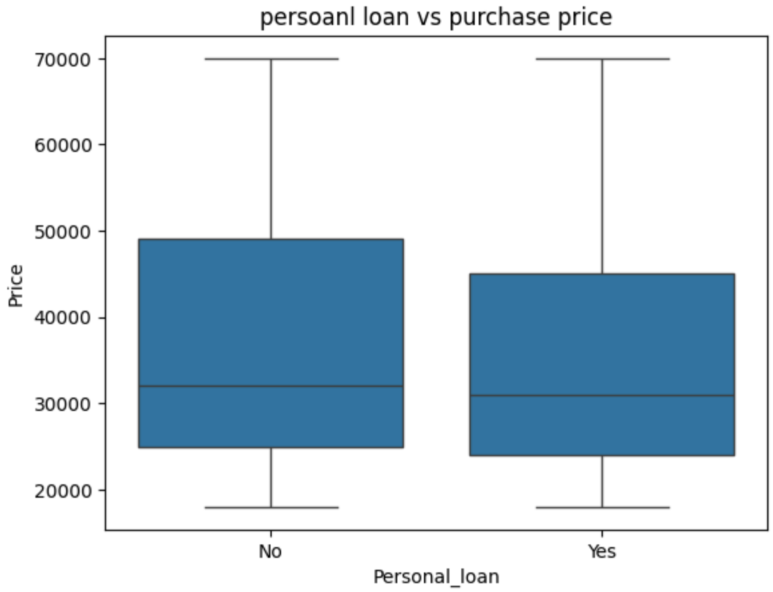


- There is a positive trend in salary and price of product
- The people with more salary can afford the cars of highest price

**personal loan vs expenditure(price)**

- understaning the expenditure pattern of those who took personal loan

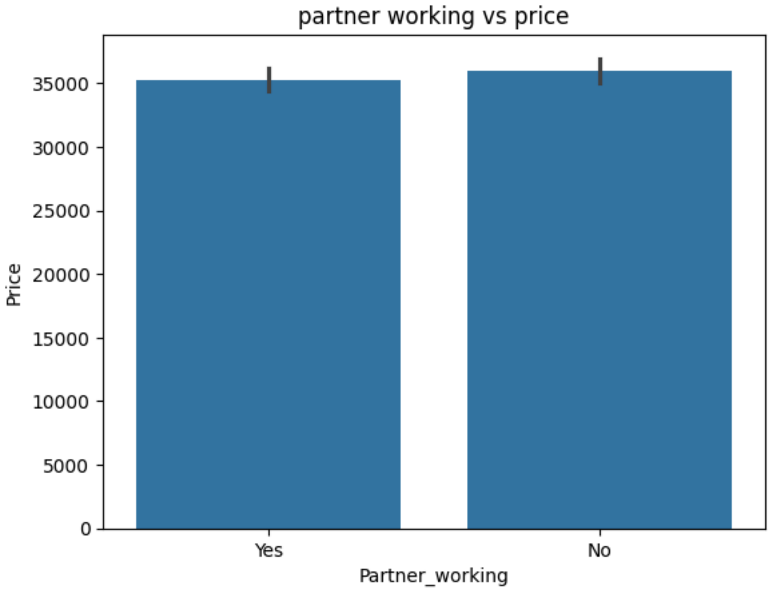
```
#boxplot
sns.boxplot(data=df,x='Personal_loan',y='Price');
plt.title('persoanl loan vs purchase price')
plt.show()
```

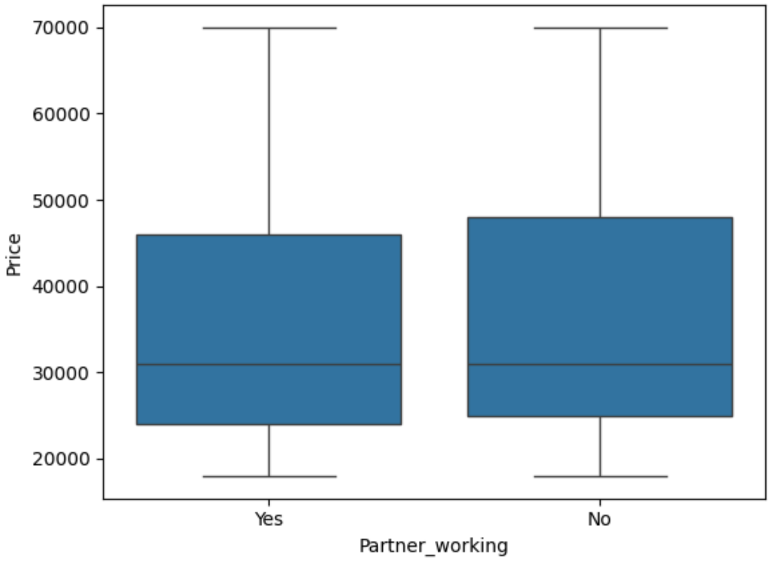


- people who are not taking loan can purchase the prouct of higher prices (slightly)

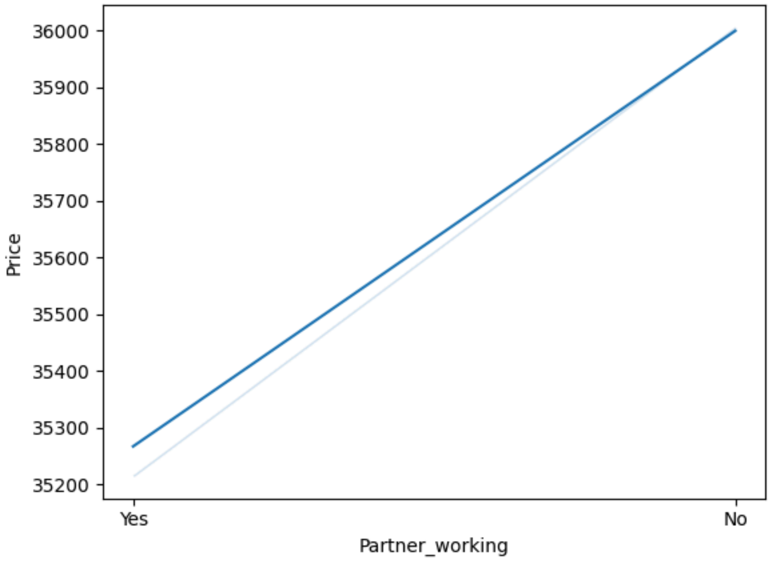
### working partner vs car price

```
sns.barplot(data=df,x='Partner_working',y='Price')
plt.title('partner working vs price')
plt.show()
sns.boxplot(data=df,x='Partner_working',y='Price')
plt.show()
sns.lineplot(data=df,x='Partner_working',y='Price',ci=False)
plt.show()
```





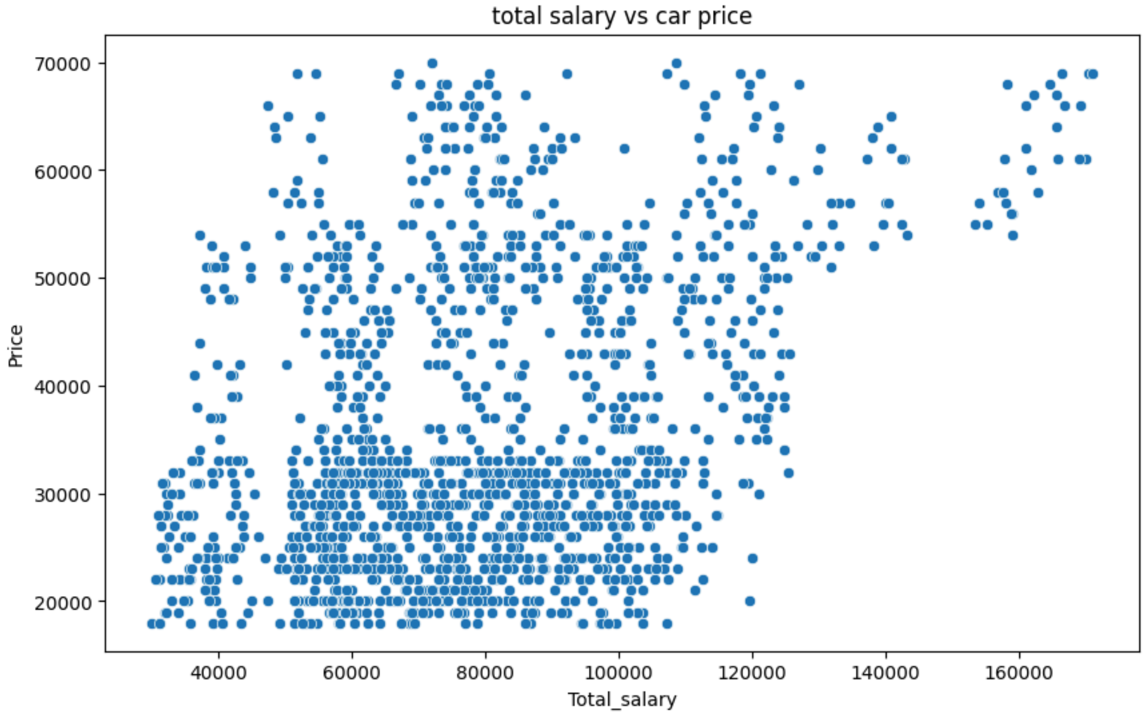
```
<ipython-input-37-29d17265f357>:6: FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', False)` for the same effect.

  sns.lineplot(data=df,x='Partner_working',y='Price',ci=False)
```



- IF the partner is also working the fiancials are more stronger and have greater purchase power(affordability) compared to the other once

**total_salary vs price**

```
plt.figure(figsize=(10,6))
sns.scatterplot(data=df,x='Total_salary',y='Price')
plt.title('total salary vs car price')
plt.show()
```

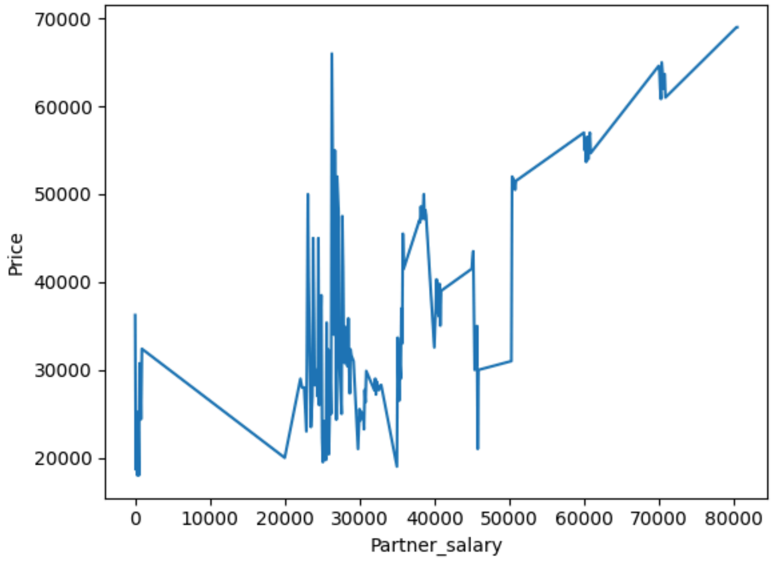


- There is a postive trend , if the total salary is more the purchase price is more so people with more salary can afford costly cars

### partner_salary vs price

```
sns.lineplot(data=df,x='Partner_salary',y='Price',ci=False)
plt.show()
```

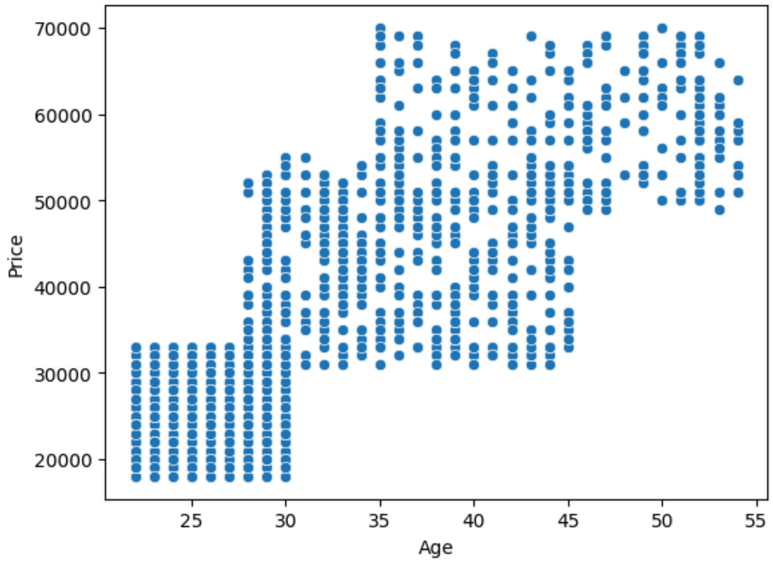
```
<ipython-input-39-9ae9720596ef>:1: FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', False)` for the same effect.

  sns.lineplot(data=df,x='Partner_salary',y='Price',ci=False)
```



- If the partners salary is more, theres more affordability

### Age vs salary

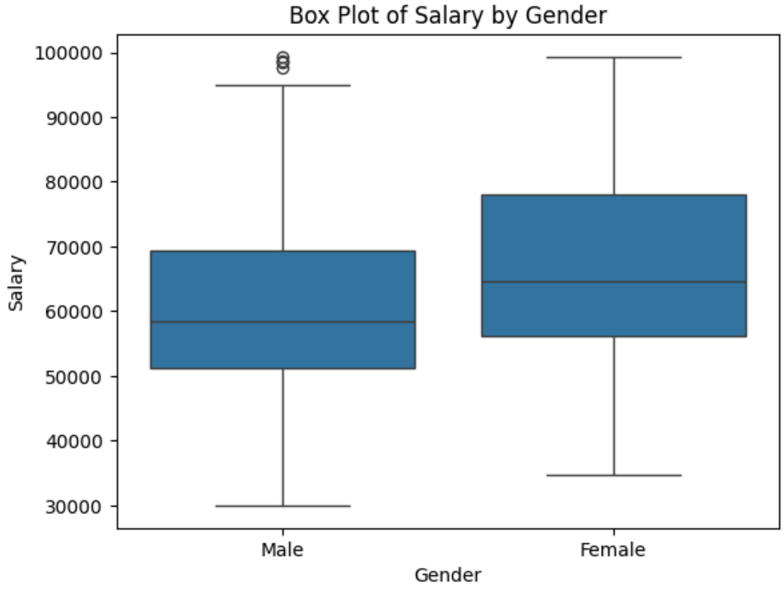
```
sns.scatterplot(data=df,x='Age',y='Price')
plt.show()
```



- as we see in the scatterplot the people with more age can afford the expensive car
- As we see that the people of age 20 to 30 can afford the car which is less expensive
- There is a positive trend in the age and price i.e affordability increases with increase in age

**Gender vs salary**

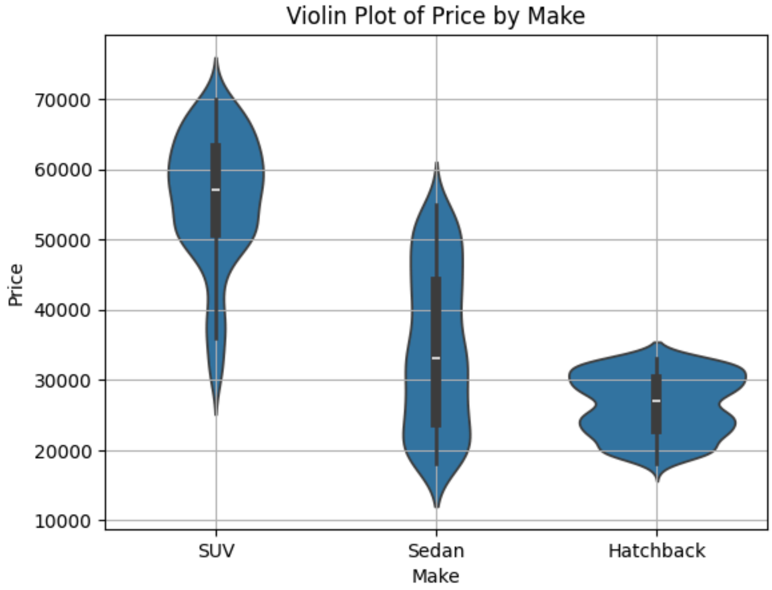
```
sns.boxplot(x='Gender', y='Salary', data=df)
plt.title('Box Plot of Salary by Gender')
plt.show()
```



- females have more salary than male
- there is few outlier in male salary
- the salary range of male is from 30k to 95k
- the salary range of female is from 35k to 100000

**make vs price**

```
sns.violinplot(x='Make', y='Price', data=df)
plt.title('Violin Plot of Price by Make')
plt.grid()
plt.show()
```

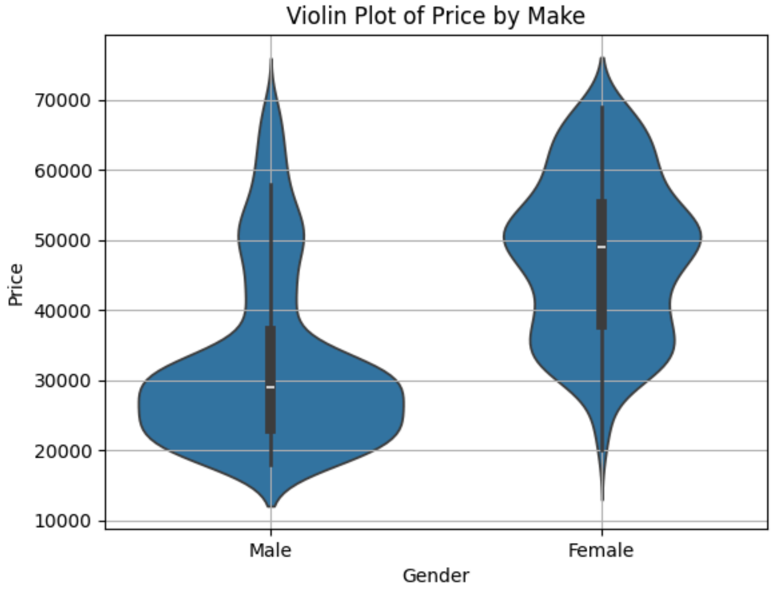


- suv: the median of suv is 55000, the peice of suv is 35000to 70000.
- sedan make: The median of sedan make price is around 33000. the price is rane is from 18000 to 54000
- hatchback: The median price of hatchback is around 26000.the price range is from 18000 to 33000

- The most expensive is suvs
- the most cheap and affordable is hatchback

**Gender vs make**

```
sns.violinplot(x='Gender', y='Price', data=df)
plt.title('Violin Plot of Price by Make')
plt.grid()
plt.show()
```



- males can afford the car ranging from 15000 to 70000 the median is 28000
- females can afford from price range of 20000 to 68000 the median is 48000

## Multivariate analysis

```
numeric_df=df.select_dtypes(include=[np.number])
corr_mat = numeric_df.corr()
```

```
corr_mat
```

| | Age | No_of_Dependents | Salary | Partner_salary | Total_salary | Price |
|---|---|---|---|---|---|---|
| **Age** | 1.000000 | -0.189614 | 0.616899 | 0.121187 | 0.458869 | 0.797831 |
| **No_of_Dependents** | -0.189614 | 1.000000 | -0.031746 | 0.121555 | 0.092890 | -0.135839 |
| **Salary** | 0.616899 | -0.031746 | 1.000000 | 0.065348 | 0.641560 | 0.409920 |
| **Partner_salary** | 0.121187 | 0.121555 | 0.065348 | 1.000000 | 0.765446 | 0.161136 |
| **Total_salary** | 0.458869 | 0.092890 | 0.641560 | 0.765446 | 1.000000 | 0.367823 |
| **Price** | 0.797831 | -0.135839 | 0.409920 | 0.161136 | 0.367823 | 1.000000 |

Next steps: View recommended plots

- Age and Price (0.798)

Interpretation: There is a strong positive correlation between Age and Price, indicating that older individuals tend to purchase more expensive automobiles. Business Insight: Marketing strategies could consider targeting older age groups for higher-priced vehicles

- Age and Salary (0.617) Interpretation: There is a moderate positive correlation between Age and Salary, suggesting that older individuals generally have higher salaries. Business Insight: The company may focus on salary brackets when designing campaigns targeting experienced, higher-income customers

- Age and Total_salary (0.459)

Interpretation: Age has a moderate positive correlation with Total Salary. This aligns with the trends in individual salary and the influence of combined household income. Business Insight: Dual-income households, especially with older individuals, can be viable targets for premium offerings.

- Salary and Price (0.410)

Interpretation: Salary has a moderate positive correlation with the Price of the automobiles purchased, indicating that higher-salary individuals tend to buy more expensive cars. Business Insight: Crafting marketing campaigns that highlight premium features and luxury targeting high-salary customers can be effective.

- Partner Salary and Total Salary (0.765)

Interpretation: There's a strong positive correlation between Partner Salary and Total Salary, which is expected as Total Salary is the sum of individual and partner salaries. Business Insight: Evaluating household income as a key factor in customer segmentation could enhance targeting strategies.

- Total Salary and Price (0.368)

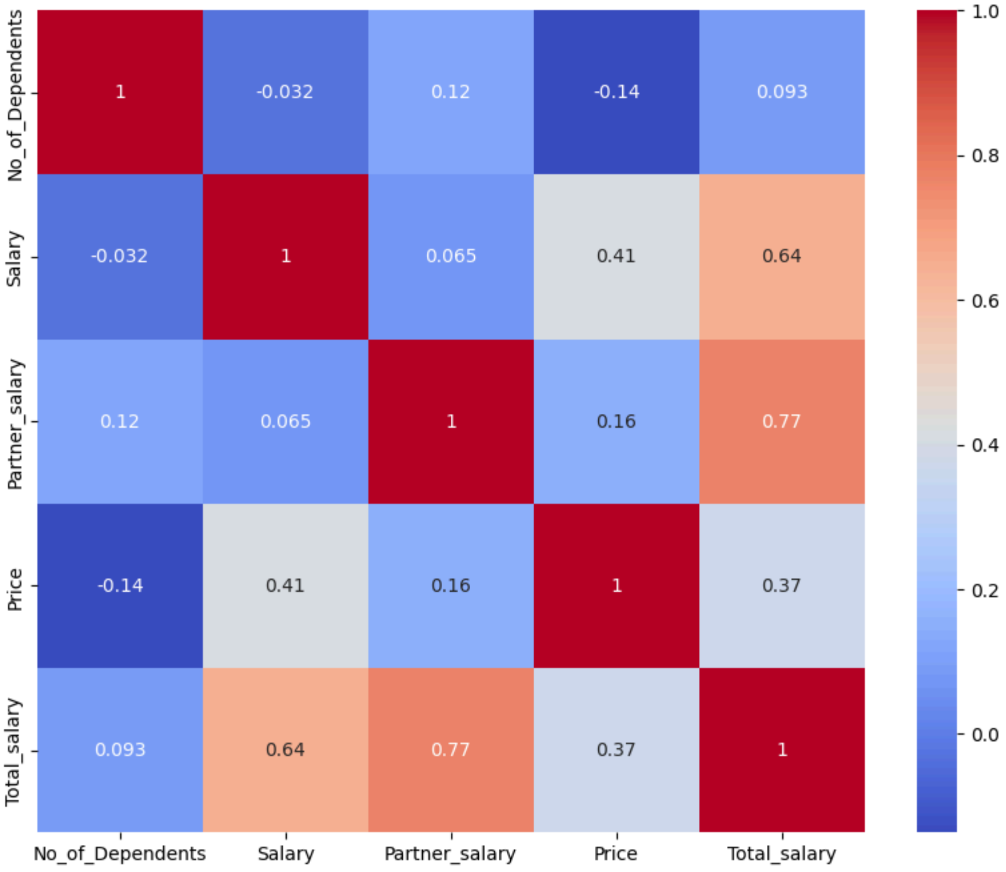
Interpretation: Total Salary has a weak positive correlation with Price, indicating that overall household income influences but does not strongly dictate automobile expenditure. Business Insight: Household income still has a role in purchasing decisions but should be considered alongside other factors like individual salaries and age.

- key takeaways Strong Correlations: Focus on Age and its relationship with Price and Salary for targeted marketing. Moderate Correlations: Recognize that Salary directly influences expenditure on automobiles. Weak Correlations: Understand that Partner Salary alone does not strongly influence the Price, but Total Salary should still be considered.

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1581 entries, 0 to 1580
Data columns (total 14 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Age               1581 non-null   int64
 1   Gender            1581 non-null   object
 2   Profession        1581 non-null   object
 3   Marital_status    1581 non-null   object
 4   Education         1581 non-null   object
 5   No_of_Dependents  1581 non-null   int64
 6   Personal_loan     1581 non-null   object
 7   House_loan        1581 non-null   object
 8   Partner_working   1581 non-null   object
 9   Salary            1581 non-null   int64
 10  Partner_salary    1581 non-null   float64
 11  Total_salary      1581 non-null   int64
 12  Price             1581 non-null   int64
 13  Make              1581 non-null   object
dtypes: float64(1), int64(5), object(8)
memory usage: 173.0+ KB
```

```
numerical_col = ['No_of_Dependents','Salary','Partner_salary','Price','Total_salary']
co_rel = df[numerical_col].corr()
plt.figure(figsize=(10, 8))
sns.heatmap(co_rel, annot=True, cmap='coolwarm')
plt.show()
```



- Total_salary and Partner_salary (0.77) This high positive correlation indicates that as the partner's salary increases, the total salary of the household also increases significantly. This is expected because Total_salary is the sum of Salary (individual's salary) and Partner_salary.
- Salary and Total_salary (0.64):

This positive correlation is also strong, indicating that as the individual's salary increases, the total household salary tends to increase. Again, this is expected because Total_salary includes the individual's Salary.

Double-click (or enter) to edit

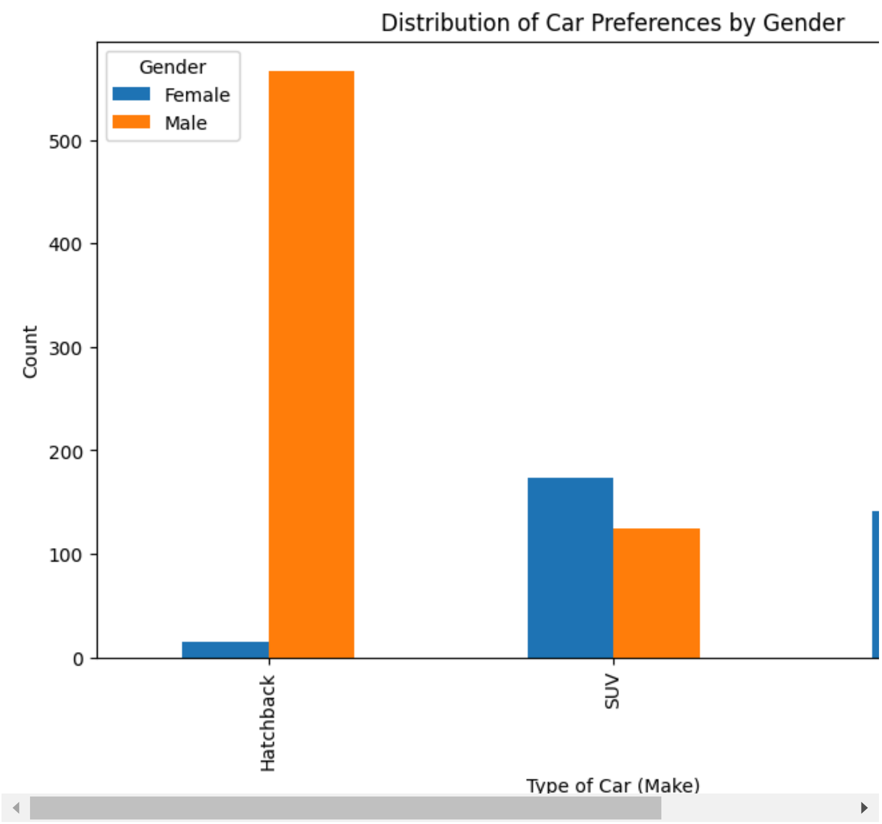
## key questions

1. Do men tend to prefer SUVs more compared to women?

```
crossrel = pd.crosstab(df['Make'],df['Gender'])
print(crossrel)
```

```
Gender     Female  Male
Make
Hatchback      15   567
SUV           173   124
Sedan         141   561
```

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

crossrel.plot(kind='bar', figsize=(10, 6))
plt.title('Distribution of Car Preferences by Gender')
plt.xlabel('Type of Car (Make)')
plt.ylabel('Count')
plt.show()
```



- as we see from above that the count of males prefering suv is 124 na dthat of female is 173 therefore , women prefer more SUVs than male

2. What is the likelihood of a salaried person buying a Sedan?

```
salaried_person = df[df['Profession'] == 'Salaried'] #filtering out the people who are salaried
total_salaried_person = salaried_person.shape[0] #getting the number count of people who are salaried
total_salaried_person
```

```
896
```

There are 896 total people who are salaried people

```
sedan_sal = salaried_person[salaried_person['Make']=='Sedan'].shape[0] #getting the number of salaried people who bought sedan
```

```
sedan_sal
```

```
396
```

- There are 396 salaried people who bought sedan

```
#getting the properotion of people
proportion = (sedan_sal/total_salaried_person)*100
```

Double-click (or enter) to edit

```
proportion
```

```
44.19642857142857
```

The likelihood of salaried person buying sedan is 44.19%

3. What evidence or data supports Sheldon Cooper's claim that a salaried male is an easier target for a SUV sale over a Sedan sale?

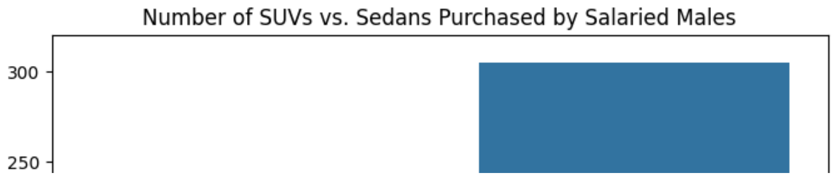
```
salaried_males = df[(df['Profession'] == 'Salaried') & (df['Gender']=='Male')] #filtering the salaried males
```

```
#getting the number of males having suvs
suv_count = salaried_males[salaried_males['Make']=='SUV'].shape[0]
#getting the number of males bought sedan
sedan_count = salaried_males[salaried_males['Make']=='Sedan'].shape[0]
suv_count
total_salaried_males = salaried_males.shape[0]
total_salaried_males
```

```
672
```

- there are 90 males who brought suvs
- There are 305 males who brought sedan
- total number of males is 672

```
vehicle_counts = pd.DataFrame({
    'Vehicle Type': ['SUV', 'Sedan'],
    'Count': [suv_count, sedan_count]
})


plt.figure(figsize=(8, 6))
sns.barplot(x='Vehicle Type', y='Count', data=vehicle_counts)
plt.xlabel('Vehicle Type')
plt.ylabel('Number of Vehicles Purchased')
plt.title('Number of SUVs vs. Sedans Purchased by Salaried Males')
plt.show()
```


Number of SUVs vs. Sedans Purchased by Salaried Males