

```
In [15]: import pandas as pd
import numpy as np
import seaborn as sns
```

```
In [16]: df=pd.read_csv("C:/Users/Rutuja/OneDrive/Desktop/fraud_oracle.csv")
```

```
In [17]: df
```

```
Out[17]:
```

	Month	WeekOfMonth	DayOfWeek	Make	AccidentArea	DayOfWeekClaimed	MonthClai
0	Dec	5	Wednesday	Honda	Urban	Tuesday	
1	Jan	3	Wednesday	Honda	Urban	Monday	
2	Oct	5	Friday	Honda	Urban	Thursday	
3	Jun	2	Saturday	Toyota	Rural	Friday	
4	Jan	5	Monday	Honda	Urban	Tuesday	
...
15415	Nov	4	Friday	Toyota	Urban	Tuesday	
15416	Nov	5	Thursday	Pontiac	Urban	Friday	
15417	Nov	5	Thursday	Toyota	Rural	Friday	
15418	Dec	1	Monday	Toyota	Urban	Thursday	
15419	Dec	2	Wednesday	Toyota	Urban	Thursday	

15420 rows × 33 columns

```
In [18]: df.columns
```

```
Out[18]: Index(['Month', 'WeekOfMonth', 'DayOfWeek', 'Make', 'AccidentArea',
       'DayOfWeekClaimed', 'MonthClaimed', 'WeekOfMonthClaimed', 'Sex',
       'MaritalStatus', 'Age', 'Fault', 'PolicyType', 'VehicleCategory',
       'VehiclePrice', 'FraudFound_P', 'PolicyNumber', 'RepNumber',
       'Deductible', 'DriverRating', 'Days_Policy_Accident',
       'Days_Policy_Claim', 'PastNumberOfClaims', 'AgeOfVehicle',
       'AgeOfPolicyHolder', 'PoliceReportFiled', 'WitnessPresent', 'AgentType',
       'NumberOfSuppliments', 'AddressChange_Claim', 'NumberOfCars', 'Year',
       'BasePolicy'],
      dtype='object')
```

```
In [5]: df['Year'].unique()
```

```
Out[5]: array([1994, 1995, 1996], dtype=int64)
```

```
In [6]: df['Year'].value_counts()
```

```
Out[6]:
```

Year	count
1994	6142
1995	5195
1996	4083

Name: count, dtype: int64

```
In [7]: df['Make'].unique()
```

```
Out[7]: array(['Honda', 'Toyota', 'Ford', 'Mazda', 'Chevrolet', 'Pontiac',
   'Accura', 'Dodge', 'Mercury', 'Jaguar', 'Nisson', 'VW', 'Saab',
   'Saturn', 'Porche', 'BMW', 'Mecedes', 'Ferrari', 'Lexus'],
  dtype=object)
```

```
In [8]: df['Make'].nunique()
```

```
Out[8]: 19
```

```
In [9]: df['Make'].value_counts()
```

```
Out[9]: Make
Pontiac      3837
Toyota       3121
Honda        2801
Mazda         2354
Chevrolet    1681
Accura        472
Ford          450
VW            283
Dodge          109
Saab           108
Mercury        83
Saturn         58
Nisson          30
BMW            15
Jaguar          6
Porche          5
Mecedes         4
Ferrari         2
Lexus            1
Name: count, dtype: int64
```

```
In [10]: df['AccidentArea'].unique()
```

```
Out[10]: array(['Urban', 'Rural'], dtype=object)
```

```
In [11]: df['AccidentArea'].value_counts()
```

```
Out[11]: AccidentArea
Urban       13822
Rural       1598
Name: count, dtype: int64
```

```
In [12]: df['Sex'].unique()
```

```
Out[12]: array(['Female', 'Male'], dtype=object)
```

```
In [13]: df['Sex'].value_counts()
```

```
Out[13]: Sex
Male        13000
Female      2420
Name: count, dtype: int64
```

```
In [14]: df['MaritalStatus'].value_counts()
```

```
Out[14]: MaritalStatus
Married     10625
Single      4684
Divorced     76
Widow        35
Name: count, dtype: int64
```

```
In [68]: df['AgeOfVehicle'].unique()
```

```
Out[68]: array(['3 years', '6 years', '7 years', 'more than 7', '5 years', 'new',  
       '4 years', '2 years'], dtype=object)
```

```
In [69]: df['AgeOfVehicle'].nunique()
```

```
Out[69]: 8
```

```
In [70]: df['AgeOfVehicle'].value_counts()
```

```
Out[70]: AgeOfVehicle  
7 years      5807  
more than 7   3981  
6 years      3448  
5 years      1357  
new          373  
4 years      229  
3 years      152  
2 years       73  
Name: count, dtype: int64
```

```
In [71]: df['AgeOfPolicyHolder'].unique()
```

```
Out[71]: array(['26 to 30', '31 to 35', '41 to 50', '51 to 65', '21 to 25',  
       '36 to 40', '16 to 17', 'over 65', '18 to 20'], dtype=object)
```

```
In [72]: df['AgeOfPolicyHolder'].nunique()
```

```
Out[72]: 9
```

```
In [73]: df['AgeOfPolicyHolder'].value_counts()
```

```
Out[73]: AgeOfPolicyHolder  
31 to 35      5593  
36 to 40      4043  
41 to 50      2828  
51 to 65      1392  
26 to 30       613  
over 65        508  
16 to 17       320  
21 to 25       108  
18 to 20        15  
Name: count, dtype: int64
```

```
In [74]: df['BasePolicy'].unique()
```

```
Out[74]: array(['Liability', 'Collision', 'All Perils'], dtype=object)
```

```
In [75]: df['BasePolicy'].value_counts()
```

```
Out[75]: BasePolicy  
Collision     5962  
Liability    5009  
All Perils    4449  
Name: count, dtype: int64
```

```
In [76]: df[(df['Sex']=='Male') & (df['AccidentArea']=='Urban')]
```

Out[76]:

	Month	WeekOfMonth	DayOfWeek	Make	AccidentArea	DayOfWeekClaimed	MonthC
1	Jan	3	Wednesday	Honda	Urban	Monday	
2	Oct	5	Friday	Honda	Urban	Thursday	
5	Oct	4	Friday	Honda	Urban	Wednesday	
6	Feb	1	Saturday	Honda	Urban	Monday	
7	Nov	1	Friday	Honda	Urban	Tuesday	
...
15413	Nov	4	Thursday	Mazda	Urban	Monday	
15414	Nov	4	Friday	Chevrolet	Urban	Monday	
15415	Nov	4	Friday	Toyota	Urban	Tuesday	
15416	Nov	5	Thursday	Pontiac	Urban	Friday	
15419	Dec	2	Wednesday	Toyota	Urban	Thursday	

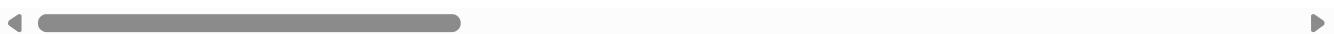
11595 rows × 33 columns

In [77]: `df[(df['Sex']=='Male') & (df['AccidentArea']=='Rural')]`

Out[77]:

	Month	WeekOfMonth	DayOfWeek	Make	AccidentArea	DayOfWeekClaimed	MonthC
3	Jun	2	Saturday	Toyota	Rural	Friday	
13	Jan	5	Friday	Honda	Rural	Wednesday	
52	Jul	3	Sunday	Honda	Rural	Wednesday	
58	May	1	Monday	Honda	Rural	Wednesday	
85	Dec	2	Tuesday	Chevrolet	Rural	Tuesday	
...
15370	Mar	5	Tuesday	Toyota	Rural	Wednesday	
15371	Mar	1	Friday	Pontiac	Rural	Friday	
15389	Jan	1	Monday	Chevrolet	Rural	Tuesday	
15411	Nov	4	Tuesday	Honda	Rural	Wednesday	
15417	Nov	5	Thursday	Toyota	Rural	Friday	

1405 rows × 33 columns

In [78]: `df[(df['Sex']=='Female') & (df['AccidentArea']=='Rural')]`

Out[78]:

	Month	WeekOfMonth	DayOfWeek	Make	AccidentArea	DayOfWeekClaimed	MonthClai
78	Feb	2	Friday	Jaguar	Rural	Tuesday	
121	Oct	2	Wednesday	Honda	Rural	Monday	
211	Feb	4	Wednesday	Honda	Rural	Monday	
236	Mar	1	Wednesday	Pontiac	Rural	Thursday	
314	Mar	5	Wednesday	Pontiac	Rural	Tuesday	
...
14700	Oct	3	Sunday	Pontiac	Rural	Tuesday	
14829	Oct	3	Friday	Pontiac	Rural	Friday	
14959	Jul	5	Monday	Pontiac	Rural	Monday	
15009	Oct	4	Friday	Mazda	Rural	Monday	
15183	Jun	1	Sunday	Honda	Rural	Monday	

193 rows × 33 columns

In [79]:

```
df[(df['Sex']=='Female') & (df['AccidentArea']=='Urban')]
```

Out[79]:

	Month	WeekOfMonth	DayOfWeek	Make	AccidentArea	DayOfWeekClaimed	MonthC
0	Dec	5	Wednesday	Honda	Urban	Tuesday	
4	Jan	5	Monday	Honda	Urban	Tuesday	
17	Jul	5	Sunday	Chevrolet	Urban	Wednesday	
29	May	3	Monday	Mazda	Urban	Wednesday	
40	Mar	4	Wednesday	Chevrolet	Urban	Wednesday	
...
15397	Jan	2	Saturday	Mazda	Urban	Tuesday	
15409	Nov	3	Friday	Mazda	Urban	Sunday	
15410	Nov	3	Sunday	Dodge	Urban	Tuesday	
15412	Nov	4	Tuesday	Pontiac	Urban	Wednesday	
15418	Dec	1	Monday	Toyota	Urban	Thursday	

2227 rows × 33 columns

In [80]:

```
data=df[(df['Sex']=='Female') & (df['AccidentArea']=='Urban')]
```

In [81]:

```
def customer_details (sex,AccidentArea,AgeOfVehicle,Year,Make):
    data=df[(df['Sex']==sex) & (df['AccidentArea']==AccidentArea) & (df['AgeOfVehicle'])
    return data;
```

In [82]:

```
customer_details ('Male','Urban','5 years',1994,'Pontiac').shape
```

```
Out[82]: (85, 33)
```

```
In [83]: customer_details ('Male','Urban','5 years',1995,'Pontiac').shape
```

```
Out[83]: (76, 33)
```

```
In [84]: customer_details ('Male','Urban','5 years',1996,'Pontiac').shape
```

```
Out[84]: (57, 33)
```

```
In [85]: customer_details ('Female','Urban','5 years',1994,'Pontiac').shape
```

```
Out[85]: (42, 33)
```

```
In [86]: customer_details ('Female','Urban','5 years',1995,'Pontiac').shape
```

```
Out[86]: (26, 33)
```

```
In [87]: customer_details ('Female','Urban','5 years',1996,'Pontiac').shape
```

```
Out[87]: (26, 33)
```

Top 5 Make

```
In [88]: df['Make'].value_counts()[0:5]
```

```
Out[88]: Make
Pontiac      3837
Toyota       3121
Honda        2801
Mazda         2354
Chevrolet    1681
Name: count, dtype: int64
```

cheaper 5 make

```
In [89]: df['Make'].value_counts().tail()
```

```
Out[89]: Make
Jaguar        6
Porche        5
Mecedes       4
Ferrari       2
Lexus          1
Name: count, dtype: int64
```

```
In [90]: df['Month'].value_counts()
```

```
Out[90]: Month
Jan      1411
May      1367
Mar      1360
Jun      1321
Oct      1305
Dec      1285
Apr      1280
Feb      1266
Jul      1257
Sep      1240
Nov      1201
Aug      1127
Name: count, dtype: int64
```

```
In [91]: df['Age'].value_counts()
```

```
Out[91]: Age
30      596
33      574
34      573
35      569
28      560
...
20      28
70      27
79      20
16      9
17      6
Name: count, Length: 66, dtype: int64
```

```
In [92]: df['NumberOfCars'].unique()
```

```
Out[92]: array(['3 to 4', '1 vehicle', '2 vehicles', '5 to 8', 'more than 8'],
              dtype=object)
```

```
In [93]: df['AgentType'].unique()
```

```
Out[93]: array(['External', 'Internal'], dtype=object)
```

```
In [94]: df['AgentType'].value_counts()
```

```
Out[94]: AgentType
External    15179
Internal     241
Name: count, dtype: int64
```

```
In [95]: df['VehicleCategory'].unique()
```

```
Out[95]: array(['Sport', 'Utility', 'Sedan'], dtype=object)
```

```
In [96]: df['VehicleCategory'].value_counts()
```

```
Out[96]: VehicleCategory
Sedan      9671
Sport      5358
Utility     391
Name: count, dtype: int64
```

In which Accident area more accident have happen by which gender type most and

what is the age of vehicle?

```
In [100...]: def accident_details (Area,gender,age):  
    data=df[(df['AccidentArea']==Area) &(df['Sex']==gender) &(df['AgeOfVehicle']==age)]  
    return data;
```

```
In [106...]: accident_details('Urban','Male','7 years').shape
```

```
Out[106]: (4414, 33)
```

```
In [102...]: accident_details('Rural','Male','7 years')
```

```
Out[102]:
```

	Month	WeekOfMonth	DayOfWeek	Make	AccidentArea	DayOfWeekClaimed	MonthC
203	Sep	1	Monday	Honda	Rural	Tuesday	
214	Jun	4	Saturday	Mazda	Rural	Monday	
228	Dec	4	Thursday	Pontiac	Rural	Wednesday	
247	Feb	3	Thursday	Accura	Rural	Friday	
292	Jan	2	Sunday	Honda	Rural	Tuesday	
...
15339	Dec	4	Wednesday	Chevrolet	Rural	Wednesday	
15370	Mar	5	Tuesday	Toyota	Rural	Wednesday	
15371	Mar	1	Friday	Pontiac	Rural	Friday	
15389	Jan	1	Monday	Chevrolet	Rural	Tuesday	
15411	Nov	4	Tuesday	Honda	Rural	Wednesday	

501 rows × 33 columns

```
In [103...]: accident_details('Urban','Female','7 years')
```

Out[103]:

	Month	WeekOfMonth	DayOfWeek	Make	AccidentArea	DayOfWeekClaimed	MonthC
17	Jul	5	Sunday	Chevrolet	Urban	Wednesday	
29	May	3	Monday	Mazda	Urban	Wednesday	
79	Feb	4	Saturday	Toyota	Urban	Friday	
125	Feb	2	Saturday	Pontiac	Urban	Monday	
150	Jan	2	Monday	Pontiac	Urban	Friday	
...
15338	Sep	3	Friday	Mazda	Urban	Monday	
15383	Dec	4	Friday	Honda	Urban	Wednesday	
15391	Jan	1	Saturday	Pontiac	Urban	Monday	
15409	Nov	3	Friday	Mazda	Urban	Sunday	
15410	Nov	3	Sunday	Dodge	Urban	Tuesday	

807 rows × 33 columns

In [104...]

accident_details('Rural','Female','7 years')

Out[104]:

	Month	WeekOfMonth	DayOfWeek	Make	AccidentArea	DayOfWeekClaimed	MonthC
236	Mar	1	Wednesday	Pontiac	Rural	Thursday	
314	Mar	5	Wednesday	Pontiac	Rural	Tuesday	
326	Nov	2	Friday	Honda	Rural	Tuesday	
565	Jul	1	Friday	Pontiac	Rural	Monday	
668	Aug	5	Sunday	Toyota	Rural	Thursday	
...
14340	Nov	1	Wednesday	Chevrolet	Rural	Thursday	
14422	Jan	2	Friday	Mazda	Rural	Saturday	
14829	Oct	3	Friday	Pontiac	Rural	Friday	
14959	Jul	5	Monday	Pontiac	Rural	Monday	
15183	Jun	1	Sunday	Honda	Rural	Monday	

85 rows × 33 columns

In [105...]

accident_details('Urban','Female','more than 7')

Out[105]:

	Month	WeekOfMonth	DayOfWeek	Make	AccidentArea	DayOfWeekClaimed	MonthC
77	Feb	2	Friday	Honda	Urban	Tuesday	
286	Jan	1	Monday	Mazda	Urban	Monday	
291	Jan	2	Saturday	Toyota	Urban	Wednesday	
356	Mar	2	Wednesday	Chevrolet	Urban	Wednesday	
420	Feb	4	Thursday	Chevrolet	Urban	Wednesday	
...
15035	Nov	4	Saturday	Pontiac	Urban	Tuesday	
15109	Jun	3	Friday	Toyota	Urban	Monday	
15206	Feb	2	Sunday	Pontiac	Urban	Wednesday	
15248	Jun	3	Wednesday	Pontiac	Urban	Tuesday	
15314	Jan	5	Saturday	Pontiac	Urban	Monday	

238 rows × 33 columns

By which make accident have happen most they use which vechicle category and which agentype they use?

In [112...]: df['Make'].value_counts()[0:5]

Out[112]:

Make	count
Pontiac	3837
Toyota	3121
Honda	2801
Mazda	2354
Chevrolet	1681

Name: count, dtype: int64

In [117...]: df['VehicleCategory'].value_counts()

Out[117]:

VehicleCategory	count
Sedan	9671
Sport	5358
Utility	391

Name: count, dtype: int64

In [119...]: df[(df['Make'] == 'Pontiac') & (df['VehicleCategory'] == 'Sedan') & (df['AgentType'] == 'A1')]

Out[119]:

	Month	WeekOfMonth	DayOfWeek	Make	AccidentArea	DayOfWeekClaimed	MonthClai
41	Mar	3	Friday	Pontiac	Urban	Wednesday	
45	Feb	2	Wednesday	Pontiac	Urban	Thursday	
49	Apr	3	Friday	Pontiac	Urban	Tuesday	
54	Jun	4	Tuesday	Pontiac	Urban	Tuesday	
64	Dec	1	Friday	Pontiac	Urban	Tuesday	
...
15385	Dec	4	Friday	Pontiac	Urban	Wednesday	
15391	Jan	1	Saturday	Pontiac	Urban	Monday	
15401	Jan	2	Tuesday	Pontiac	Urban	Tuesday	
15403	Nov	2	Sunday	Pontiac	Urban	Monday	
15405	Nov	3	Wednesday	Pontiac	Urban	Tuesday	

2542 rows × 33 columns

In [122...]

```
df[(df['Make']=='Pontiac') & (df['VehicleCategory']=='Sedan') & (df['AgentType']=='
```

Out[122]:

	Month	WeekOfMonth	DayOfWeek	Make	AccidentArea	DayOfWeekClaimed	MonthClai
46	Feb	2	Thursday	Pontiac	Urban	Monday	
1116	Aug	4	Friday	Pontiac	Urban	Monday	
1473	Nov	4	Sunday	Pontiac	Urban	Monday	
1854	Jan	1	Monday	Pontiac	Urban	Friday	
2501	May	1	Monday	Pontiac	Urban	Thursday	
3068	Dec	5	Wednesday	Pontiac	Urban	Monday	
3182	Sep	5	Friday	Pontiac	Rural	Wednesday	
3778	Oct	1	Sunday	Pontiac	Rural	Thursday	
4151	Apr	4	Saturday	Pontiac	Urban	Monday	
5643	Oct	1	Sunday	Pontiac	Urban	Monday	
6500	Dec	2	Monday	Pontiac	Urban	Wednesday	
6828	Feb	3	Wednesday	Pontiac	Rural	Friday	
7489	Jul	2	Saturday	Pontiac	Urban	Tuesday	
8493	Feb	1	Friday	Pontiac	Urban	Friday	
10580	Oct	3	Sunday	Pontiac	Urban	Tuesday	
10698	Mar	2	Thursday	Pontiac	Urban	Tuesday	
10807	Oct	1	Thursday	Pontiac	Urban	Friday	
12747	Jun	1	Saturday	Pontiac	Urban	Friday	
13250	Jul	1	Saturday	Pontiac	Urban	Tuesday	
13967	Aug	1	Thursday	Pontiac	Urban	Friday	
14053	Nov	1	Saturday	Pontiac	Rural	Tuesday	
14265	Nov	4	Monday	Pontiac	Urban	Monday	

22 rows × 33 columns

In [123...]

```
df[(df['Make'] == 'Toyota') & (df['VehicleCategory'] == 'Sedan') & (df['AgentType'] == 'E
```

Out[123]:

	Month	WeekOfMonth	DayOfWeek	Make	AccidentArea	DayOfWeekClaimed	MonthClaimed
39	Nov	4	Thursday	Toyota	Urban	Tuesday	
44	Jan	3	Wednesday	Toyota	Urban	Wednesday	
60	Nov	1	Monday	Toyota	Urban	Monday	
68	Jan	1	Sunday	Toyota	Urban	Tuesday	
79	Feb	4	Saturday	Toyota	Urban	Friday	
...
15408	Nov	3	Thursday	Toyota	Urban	Wednesday	
15415	Nov	4	Friday	Toyota	Urban	Tuesday	
15417	Nov	5	Thursday	Toyota	Rural	Friday	
15418	Dec	1	Monday	Toyota	Urban	Thursday	
15419	Dec	2	Wednesday	Toyota	Urban	Thursday	

2121 rows × 33 columns

In [124...]

```
df[(df['Make']=='Toyota') & (df['VehicleCategory']=='Sedan') & (df['AgentType']=='']
```

Out[124]:

	Month	WeekOfMonth	DayOfWeek	Make	AccidentArea	DayOfWeekClaimed	MonthClaimed
688	May	3	Tuesday	Toyota	Urban	Tuesday	
836	Jul	1	Monday	Toyota	Urban	Tuesday	
1050	Mar	2	Monday	Toyota	Urban	Saturday	
2445	Jun	4	Wednesday	Toyota	Urban	Friday	
2507	Sep	3	Wednesday	Toyota	Urban	Thursday	
3308	May	4	Wednesday	Toyota	Urban	Thursday	
4071	Oct	4	Sunday	Toyota	Urban	Monday	
5004	Sep	4	Tuesday	Toyota	Urban	Monday	
6496	Jul	2	Wednesday	Toyota	Urban	Wednesday	
6601	Sep	4	Friday	Toyota	Urban	Friday	
7462	Sep	2	Thursday	Toyota	Urban	Tuesday	
7544	Jul	4	Friday	Toyota	Urban	Thursday	
8137	Sep	1	Friday	Toyota	Urban	Friday	
8148	Apr	5	Sunday	Toyota	Urban	Wednesday	
8269	Dec	2	Monday	Toyota	Urban	Thursday	
9864	Sep	2	Tuesday	Toyota	Urban	Thursday	
10121	Aug	5	Tuesday	Toyota	Urban	Friday	
10177	Jan	2	Wednesday	Toyota	Urban	Wednesday	
10666	Feb	4	Saturday	Toyota	Urban	Friday	
11171	Aug	3	Thursday	Toyota	Urban	Thursday	
11484	Dec	3	Friday	Toyota	Urban	Monday	
11775	Jun	1	Thursday	Toyota	Urban	Friday	
12103	May	2	Thursday	Toyota	Urban	Friday	
13772	Jan	2	Tuesday	Toyota	Urban	Wednesday	
14194	Apr	5	Sunday	Toyota	Rural	Wednesday	
14660	Aug	4	Sunday	Toyota	Rural	Friday	
15108	Jun	4	Wednesday	Toyota	Urban	Friday	

27 rows × 33 columns

Which car sell most then what is the price of that and car is sell less then what is the price for that?

In [126...]

df['VehiclePrice'].unique()

```
Out[126]: array(['more than 69000', '20000 to 29000', '30000 to 39000',
   'less than 20000', '40000 to 59000', '60000 to 69000'],
  dtype=object)
```

```
In [139...]: def price(car):
    data= df[df['Make'] == car]
    return data['VehiclePrice']
```

```
In [142...]: price('Pontiac').value_counts()
```

```
Out[142]: VehiclePrice
20000 to 29000    2263
30000 to 39000    884
less than 20000    436
more than 69000    187
40000 to 59000     65
60000 to 69000      2
Name: count, dtype: int64
```

```
In [141...]: price('Toyota').value_counts()
```

```
Out[141]: VehiclePrice
20000 to 29000    1958
30000 to 39000    530
less than 20000    405
more than 69000    208
40000 to 59000     19
60000 to 69000      1
Name: count, dtype: int64
```

```
In [143...]: price('Jaguar').value_counts()
```

```
Out[143]: VehiclePrice
30000 to 39000     3
40000 to 59000     3
Name: count, dtype: int64
```

```
In [144...]: price('Porche').value_counts()
```

```
Out[144]: VehiclePrice
more than 69000     3
40000 to 59000     2
Name: count, dtype: int64
```

```
In [145...]: price('Ferrari').value_counts()
```

```
Out[145]: VehiclePrice
40000 to 59000     2
Name: count, dtype: int64
```

```
In [146...]: df['PolicyType'].value_counts()
```

```
Out[146]: PolicyType
Sedan - Collision    5584
Sedan - Liability    4987
Sedan - All Perils    4087
Sport - Collision     348
Utility - All Perils   340
Utility - Collision     30
Sport - All Perils      22
Utility - Liability      21
Sport - Liability        1
Name: count, dtype: int64
```

```
In [149]: def price(cat):
    data= df[df['VehicleCategory'] == cat]
    return data['VehiclePrice']
```

```
In [151]: price('Sedan').value_counts()
```

```
Out[151]: VehiclePrice
20000 to 29000    5435
30000 to 39000    2044
less than 20000    1010
more than 69000    903
40000 to 59000    237
60000 to 69000     42
Name: count, dtype: int64
```

```
In [153]: price('Sport').value_counts()
```

```
Out[153]: VehiclePrice
20000 to 29000    2643
30000 to 39000    1480
more than 69000    923
40000 to 59000    183
less than 20000     86
60000 to 69000     43
Name: count, dtype: int64
```

```
In [154]: price('Utility').value_counts()
```

```
Out[154]: VehiclePrice
more than 69000    338
40000 to 59000     41
30000 to 39000      9
60000 to 69000      2
20000 to 29000      1
Name: count, dtype: int64
```

```
In [5]: df['Fault'].value_counts()
```

```
Out[5]: Fault
Policy Holder     11230
Third Party       4190
Name: count, dtype: int64
```

```
In [6]: df['DayOfWeek'].value_counts()
```

```
Out[6]: DayOfWeek
Monday           2616
Friday            2445
Tuesday          2300
Thursday         2173
Wednesday        2159
Saturday          1982
Sunday            1745
Name: count, dtype: int64
```

```
In [11]: def Accidenttype(Area,week,fault):
    data= df[(df['AccidentArea']==Area) & (df['DayOfWeek']==week) & (df['Fault']==fault)]
    return data
```

```
In [12]: Accidenttype('Urban','Monday','Policy Holder')
```

Out[12]:

	Month	WeekOfMonth	DayOfWeek	Make	AccidentArea	DayOfWeekClaimed	MonthClai
11	Mar	5	Monday	Honda	Urban	Monday	
14	Jan	5	Monday	Ford	Urban	Thursday	
25	Mar	3	Monday	Pontiac	Urban	Tuesday	
29	May	3	Monday	Mazda	Urban	Wednesday	
42	Jul	1	Monday	Toyota	Urban	Monday	
...
15378	Jan	2	Monday	Toyota	Urban	Friday	
15386	Dec	4	Monday	Toyota	Urban	Monday	
15390	Jan	1	Monday	Mazda	Urban	Wednesday	
15402	Jan	2	Monday	Toyota	Urban	Monday	
15404	Nov	2	Monday	Toyota	Urban	Friday	

1775 rows × 33 columns



In [13]: Accidenttype('Urban', 'Monday', 'Third Party')

Out[13]:

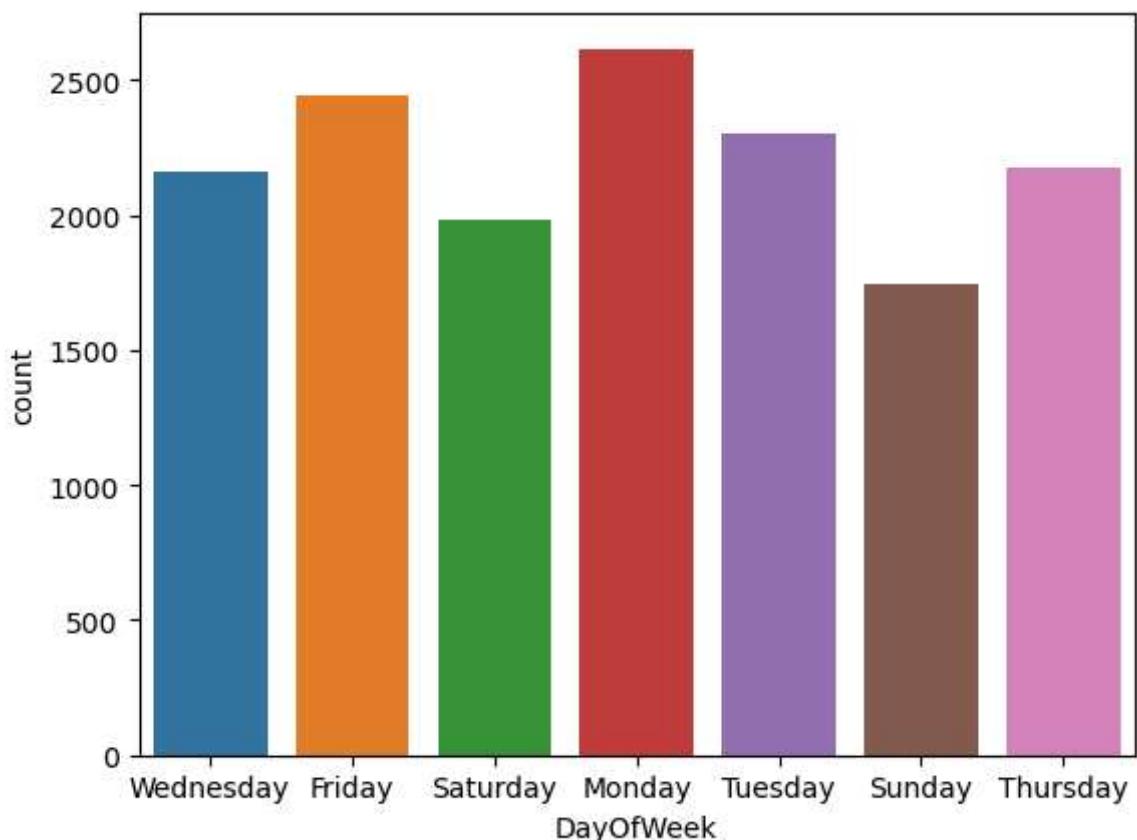
	Month	WeekOfMonth	DayOfWeek	Make	AccidentArea	DayOfWeekClaimed	MonthC
4	Jan	5	Monday	Honda	Urban	Tuesday	
19	Apr	4	Monday	Honda	Urban	Tuesday	
33	Jan	4	Monday	Honda	Urban	Tuesday	
55	Dec	3	Monday	Mazda	Urban	Wednesday	
73	Jan	1	Monday	Honda	Urban	Tuesday	
...
15319	Mar	4	Monday	Pontiac	Urban	Tuesday	
15337	Oct	2	Monday	Honda	Urban	Thursday	
15341	Nov	1	Monday	Honda	Urban	Tuesday	
15384	Dec	4	Monday	Chevrolet	Urban	Tuesday	
15418	Dec	1	Monday	Toyota	Urban	Thursday	

584 rows × 33 columns



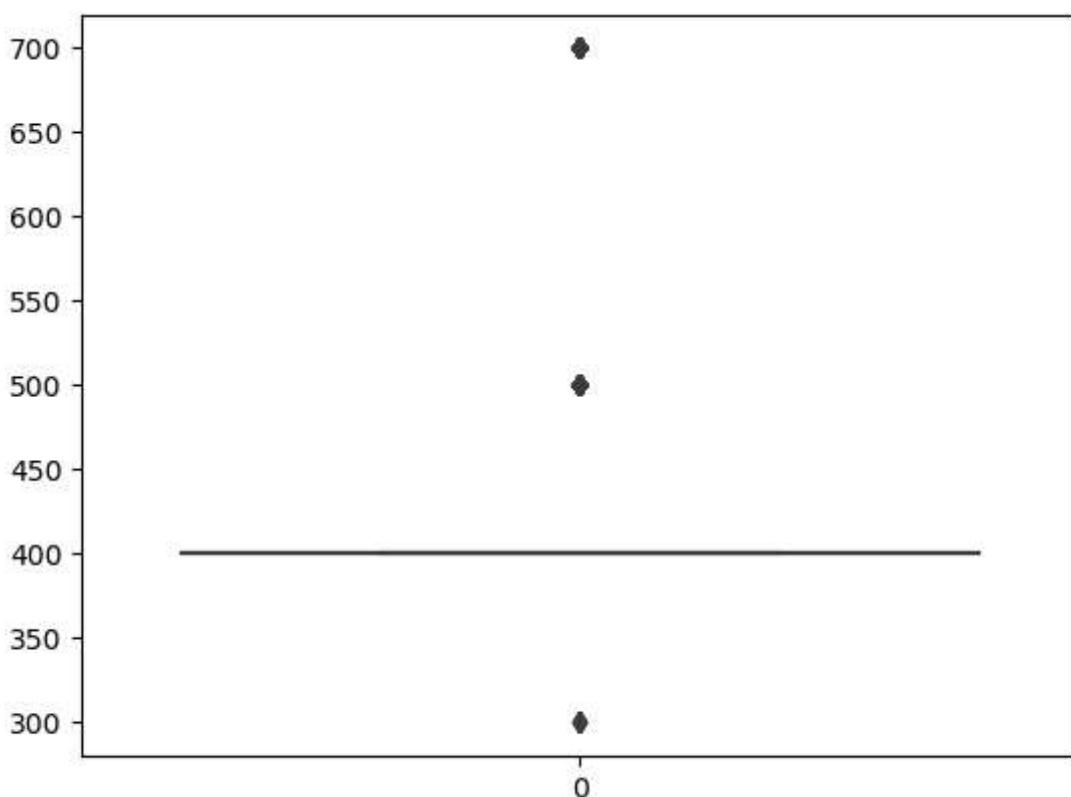
In [24]: sns.countplot(df, x=df['DayOfWeek'])

Out[24]: <Axes: xlabel='DayOfWeek', ylabel='count'>



```
In [26]: sns.boxplot(df['Deductible'])
```

```
Out[26]: <Axes: >
```



```
In [27]: pd.crosstab(df['AccidentArea'], df['VehicleCategory'])
```

Out[27]: `VehicleCategory Sedan Sport Utility`

AccidentArea		Sedan	Sport	Utility
	Rural	1129	413	56
	Urban	8542	4945	335

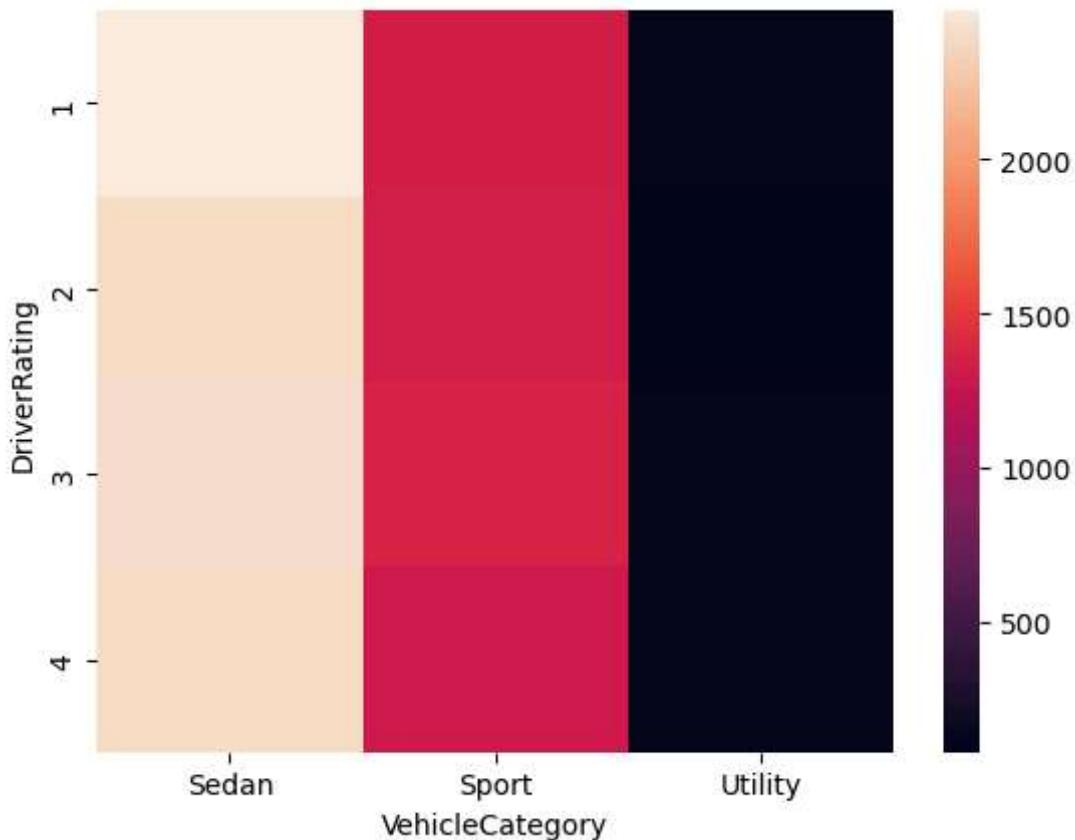
In [28]: `pd.crosstab(df['Make'], df['VehicleCategory'])`

Out[28]: `VehicleCategory Sedan Sport Utility`

Make		Sedan	Sport	Utility
	Accura	298	97	77
	BMW	9	6	0
	Chevrolet	971	579	131
	Dodge	67	42	0
	Ferrari	0	0	2
	Ford	283	131	36
	Honda	1570	1222	9
	Jaguar	3	3	0
	Lexus	1	0	0
	Mazda	1384	935	35
	Mecedes	4	0	0
	Mercury	58	17	8
	Nisson	18	6	6
	Pontiac	2564	1191	82
	Porche	0	2	3
	Saab	81	27	0
	Saturn	48	10	0
	Toyota	2148	972	1
	VW	164	118	1

In [32]: `sns.heatmap(pd.crosstab(df['DriverRating'], df['VehicleCategory']))`

Out[32]: `<Axes: xlabel='VehicleCategory', ylabel='DriverRating'>`



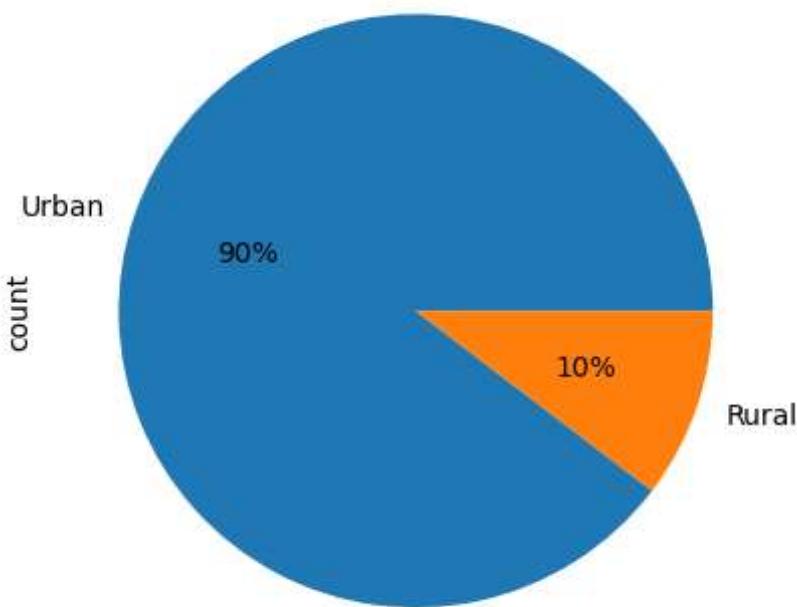
```
In [31]: pd.crosstab(df['Days_Policy_Claim'], df['Days_Policy_Accident'])
```

```
Out[31]: Days_Policy_Accident 1 to 7 15 to 30 8 to 15 more than 30 none
```

Days_Policy_Claim					
		15 to 30	8 to 15	more than 30	none
15 to 30	15 to 30	6	16	20	3
8 to 15	8 to 15	1	0	13	0
more than 30	more than 30	7	33	22	15243
none	none	0	0	0	37

```
In [69]: df['AccidentArea'].value_counts().plot(kind='pie', autopct="%01.f%%")
```

```
Out[69]: <Axes: ylabel='count'>
```



```
In [70]: pd.crosstab(df['VehicleCategory'], df['VehiclePrice'])
```

VehiclePrice	20000 to 29000	30000 to 39000	40000 to 59000	60000 to 69000	less than 20000	more than 69000
VehicleCategory						
Sedan	5435	2044	237	42	1010	903
Sport	2643	1480	183	43	86	923
Utility	1	9	41	2	0	338

```
In [73]: pd.crosstab(df['VehicleCategory'], df['Make'])
```

Make	Accura	BMW	Chevrolet	Dodge	Ferrari	Ford	Honda	Jaguar	Lexus	Mazda
VehicleCategory										
Sedan	298	9	971	67	0	283	1570	3	1	1384
Sport	97	6	579	42	0	131	1222	3	0	935
Utility	77	0	131	0	2	36	9	0	0	35

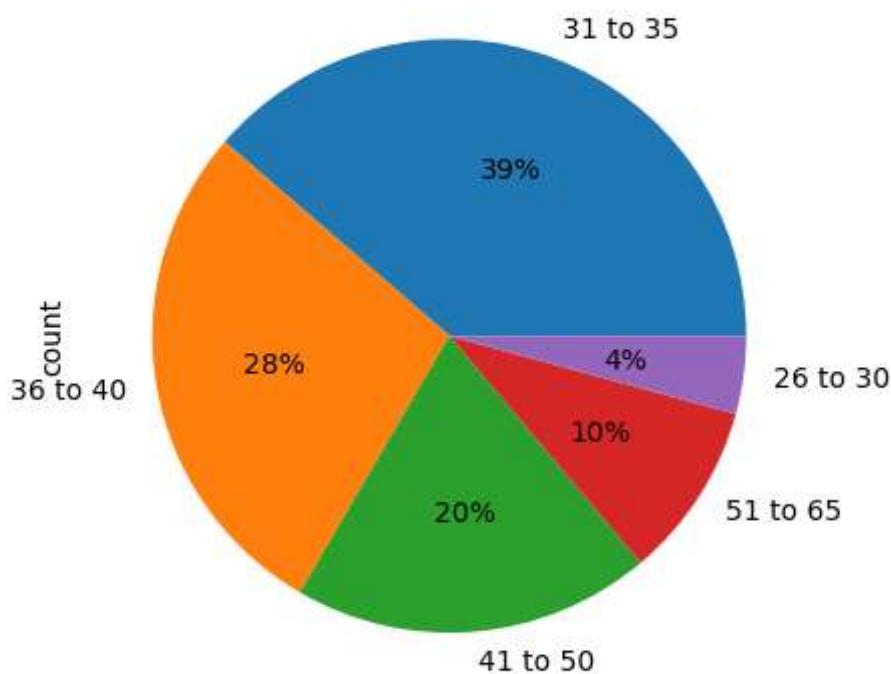
```
In [74]: pd.crosstab(df['Fault'], df['Age'])
```

	Age	0	16	17	18	19	20	21	22	23	24	...	71	72	73	74	75	76	77	78	79	80
	Fault																					
Policy Holder	278	2	5	38	24	17	82	74	82	100	...	30	33	24	25	28	33	22	27	17	27	
Third Party	42	7	1	10	8	11	45	51	40	35	...	10	12	8	10	6	9	7	8	3	5	

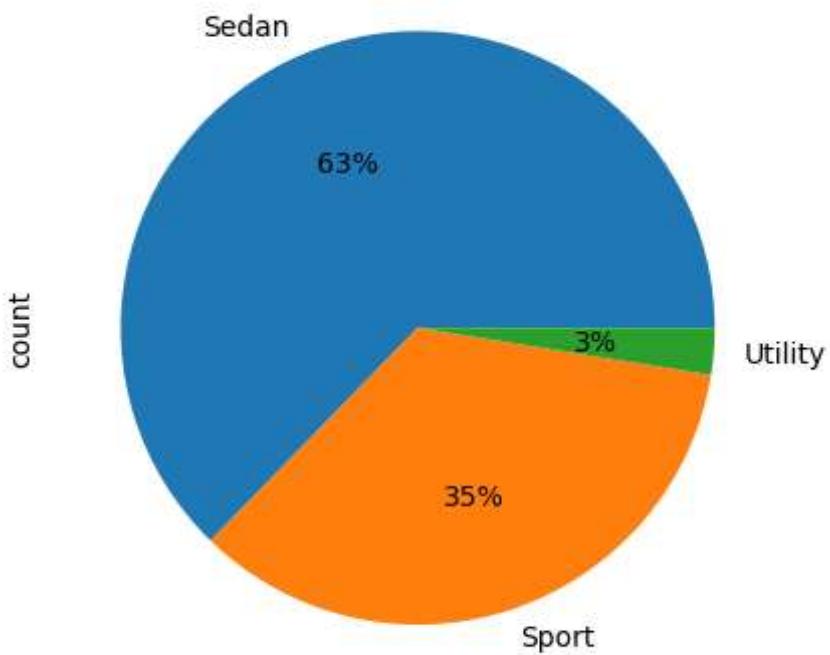
2 rows × 66 columns

In [78]: `df['Age'].unique()`Out[78]: `array([21, 34, 47, 65, 27, 20, 36, 0, 30, 42, 71, 52, 28, 61, 38, 41, 32, 40, 63, 31, 45, 60, 39, 55, 35, 44, 72, 29, 37, 59, 49, 50, 26, 48, 64, 33, 74, 23, 25, 56, 16, 68, 18, 51, 22, 53, 46, 43, 57, 54, 69, 67, 19, 78, 77, 75, 80, 58, 73, 24, 76, 62, 79, 70, 17, 66], dtype=int64)`In [83]: `df['AgeOfPolicyHolder'].value_counts()[0:5].plot(kind='pie', autopct="%01.f%%")`

Out[83]: <Axes: ylabel='count'>

In [85]: `df['VehicleCategory'].value_counts().plot(kind='pie', autopct="%01.f%%")`

Out[85]: <Axes: ylabel='count'>



In []: