In [1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

In [2]:
```
df = pd.read_csv('car_ownerships.csv')
```

In [3]:
```
df.head()
```

Out[3]:

	cars_id	Date	Sales_person	Customer_name	Car_Make	Car_Model	Car_Year	Sale_Price	Commission_Ra
0	1	01/08/2022	Monica Moore MD	Mary Butler	Nissan	Altima	2018	15983	0.07049
1	2	15/03/2023	Roberto Rose	Richard Pierce	Nissan	F-150	2016	38474	0.13443
2	3	29/04/2023	Ashley Ramos	Sandra Moore	Ford	Civic	2016	33340	0.11453
3	4	04/09/2022	Patrick Harris	Johnny Scott	Ford	Altima	2013	41937	0.09219
4	5	16/06/2022	Eric Lopez	Vanessa Jones	Honda	Silverado	2022	20256	0.11349

In [4]:
```
df.tail()
```

Out[4]:

	cars_id	Date	Sales_person	Customer_name	Car_Make	Car_Model	Car_Year	Sale_Price	Commi
1048570	1048571	22/03/2023	Jean Stevenson MD	Jeffrey Hernandez	Ford	Silverado	2015	41043	
1048571	1048572	12/02/2023	Daniel Odonnell	Amy Flores	Ford	Civic	2010	23958	
1048572	1048573	17/10/2022	Paul Foster	Lucas Frank	Ford	Corolla	2021	43768	
1048573	1048574	21/12/2022	Brett Hansen	Natasha Payne	Nissan	Altima	2010	33733	
1048574	1048575	19/02/2023	Benjamin Lawson	Daniel Hayes	Chevrolet	F-150	2019	19352	

In [5]:
```
df.shape
```

Out[5]:
```
(1048575, 10)
```

In [6]:
```
df.info
```

Out[6]:
```
<bound method DataFrame.info of          cars_id        Date        Sales_person      Cus
tomer_name   Car_Make  \
0              1  01/08/2022     Monica Moore MD         Mary Butler     Nissan
1              2  15/03/2023        Roberto Rose      Richard Pierce     Nissan
2              3  29/04/2023        Ashley Ramos        Sandra Moore       Ford
3              4  04/09/2022      Patrick Harris        Johnny Scott       Ford
4              5  16/06/2022          Eric Lopez       Vanessa Jones      Honda
...          ...         ...                 ...                 ...        ...
1048570  1048571  22/03/2023  Jean Stevenson MD   Jeffrey Hernandez       Ford
1048571  1048572  12/02/2023     Daniel Odonnell          Amy Flores       Ford
1048572  1048573  17/10/2022         Paul Foster         Lucas Frank       Ford
1048573  1048574  21/12/2022        Brett Hansen       Natasha Payne     Nissan
1048574  1048575  19/02/2023     Benjamin Lawson         Daniel Hayes  Chevrolet

          Car_Model  Car_Year  Sale_Price  Commission_Rate  Commission_Earned
```

```
0          Altima    2018     15983       0.070495         1126.73
1           F-150    2016     38474       0.134439         5172.40
2           Civic    2016     33340       0.114536         3818.63
3          Altima    2013     41937       0.092191         3866.20
4       Silverado    2022     20256       0.113490         2298.85
...           ...     ...       ...            ...             ...
1048570 Silverado    2015     41043       0.107835         4425.89
1048571     Civic    2010     23958       0.067569         1618.81
1048572   Corolla    2021     43768       0.138509         6062.26
1048573    Altima    2010     33733       0.136388         4600.77
1048574     F-150    2019     19352       0.126393         2445.96

[1048575 rows x 10 columns]>
```

In [7]:
```
df.describe()
```

Out[7]:

	cars_id	Car_Year	Sale_Price	Commission_Rate	Commission_Earned
count	1.048575e+06	1.048575e+06	1.048575e+06	1.048575e+06	1.048575e+06
mean	5.242880e+05	2.015995e+03	3.001837e+04	9.995861e-02	3.000824e+03
std	3.026977e+05	3.740360e+00	1.154381e+04	2.886096e-02	1.480912e+03
min	1.000000e+00	2.010000e+03	1.000000e+04	5.000014e-02	5.013400e+02
25%	2.621445e+05	2.013000e+03	2.003000e+04	7.493990e-02	1.822280e+03
50%	5.242880e+05	2.016000e+03	3.000900e+04	9.996744e-02	2.741750e+03
75%	7.864315e+05	2.019000e+03	4.003200e+04	1.249475e-01	3.978925e+03
max	1.048575e+06	2.022000e+03	5.000000e+04	1.500000e-01	7.494530e+03

In [8]:
```
df.columns
```

Out[8]:
```
Index(['cars_id', 'Date', 'Sales_person', 'Customer_name', 'Car_Make',
       'Car_Model', 'Car_Year', 'Sale_Price', 'Commission_Rate',
       'Commission_Earned'],
      dtype='object')
```

Check Missing Values

In [36]:
```
df.isnull().sum()
```

Out[36]:
```
cars_id              0
Date                 0
Sales_person         0
Customer_name        0
Car_Make             0
Car_Model            0
Car_Year             0
Sale_Price           0
Commission_Rate      0
Commission_Earned    0
dtype: int64
```

In [9]:
```
df.sort_values('Car_Year', ascending = False)
```

Out[9]:

	cars_id	Date	Sales_person	Customer_name	Car_Make	Car_Model	Car_Year	Sale_Price	Commissi
524287	524288	04/11/2022	Timothy Sandoval	Mr. Mark Baker	Ford	Corolla	2022	21695	(
398312	398313	07/04/2023	Jerry Hobbs	William Gonzalez	Nissan	Altima	2022	39306	(
876103	876104	18/08/2022	Christina	Brenda	Toyota	Silverado	2022	47851	(

			Carroll	Richardson					
504247	504248	01/06/2022	Tina Salas	Heather Garcia	Ford	F-150	2022	26727	(
398303	398304	15/10/2022	Amanda Roberson	Vanessa Marquez	Ford	Civic	2022	31190	(
...	...	...	...	...	...	...	...	...	
373325	373326	05/12/2022	Frederick Mccall	Adam Huerta	Ford	Altima	2010	48356	(
944857	944858	02/01/2023	Roy Harris	April Gallegos	Toyota	Civic	2010	20443	(
41318	41319	05/05/2022	Cassandra Gilbert	Colleen Davis	Ford	Altima	2010	37330	(
784431	784432	11/02/2023	Hannah Lawson	James Shaw	Chevrolet	Corolla	2010	40945	(
222246	222247	19/11/2022	Matthew Anderson II	Catherine Hoover	Nissan	Civic	2010	18626	(

1048575 rows × 10 columns

Calculate the average sale price for each car make

In [37]:

```
avg_sale_price = df.groupby('Car_Make')['Sale_Price'].mean()
```

Define a higher price of car make as the car make with the highest average sale price

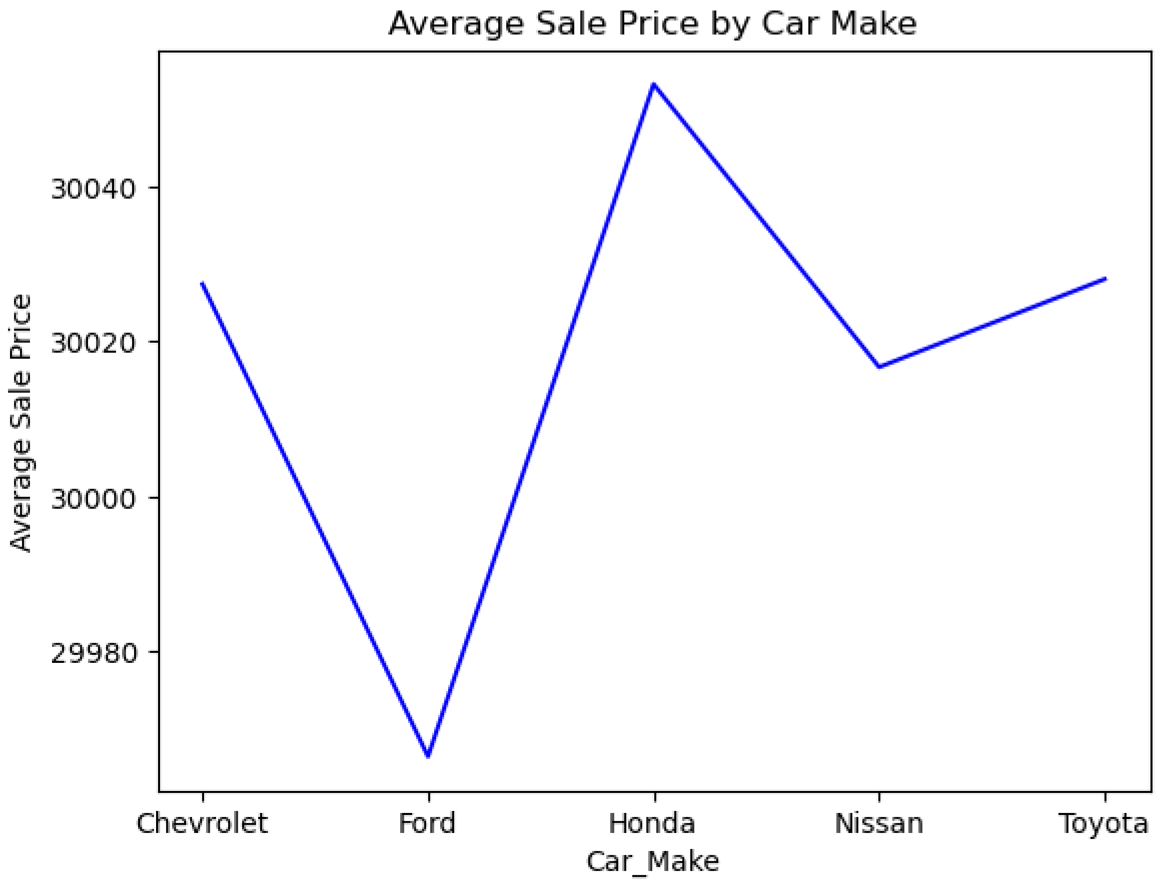
In [39]:

```
higher_price_make = avg_sale_price.idxmax()
```

Visualise the results

In [40]:

```
plt.plot(avg_sale_price, 'b-', label='Average Sale Price')
plt.xlabel('Car_Make')
plt.ylabel('Average Sale Price')
plt.title('Average Sale Price by Car Make')
plt.show()
```



In [41]:

```
print(f'The higher price of car make is {higher_price_make}')
```

```
The higher price of car make is Honda
```

In [12]:

```
unique_car_makes = df['Car_Make'].unique()
```

In [13]:

```
unique_car_makes
```

Out[13]:

```
array(['Nissan', 'Ford', 'Honda', 'Toyota', 'Chevrolet'], dtype=object)
```

In [14]:

```
unique_car_models = df['Car_Model'].unique()
```

In [15]:

```
unique_car_models
```

Out[15]:

```
array(['Altima', 'F-150', 'Civic', 'Silverado', 'Corolla'], dtype=object)
```

In [17]:

```
car_make_counts = df['Car_Make'].value_counts()
print(car_make_counts)
```

```
Car_Make
Honda        210101
Ford         210074
Chevrolet    209971
Toyota       209315
Nissan       209114
Name: count, dtype: int64
```

In [18]:

```
car_model_counts = df['Car_Model'].value_counts()
print(car_model_counts)
```

```
Car_Model
Silverado    210257
Civic        209918
F-150        209643
Corolla      209422
```

```
Altima       209335
Name: count, dtype: int64
```

In [19]:

```
car_model_perc = df['Car_Model'].value_counts(normalize=True) * 100
print(car_model_perc)

car_make_perc = df['Car_Make'].value_counts(normalize=True) * 100
print(car_make_perc)
```

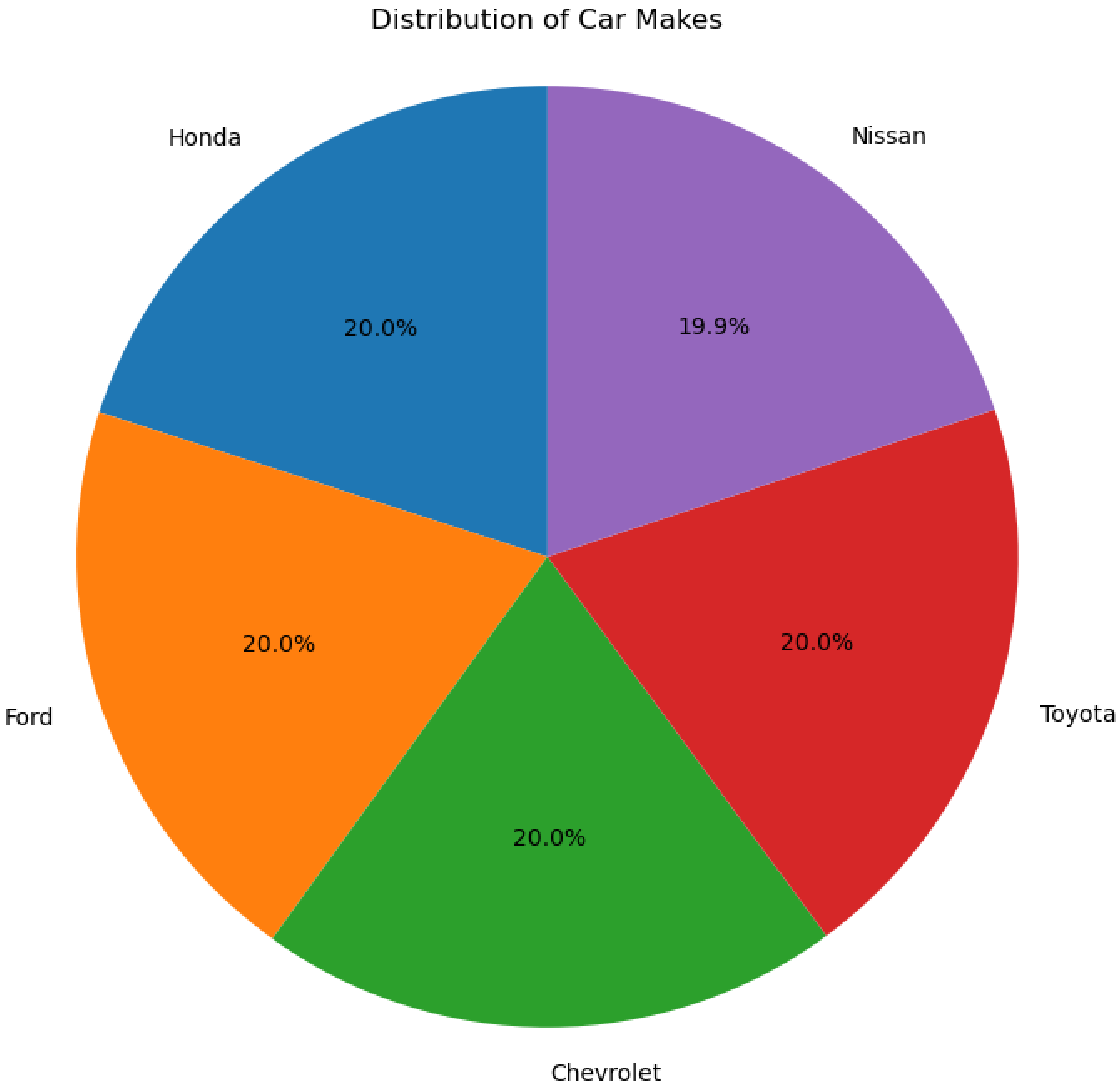
```
Car_Model
Silverado    20.051689
Civic        20.019360
F-150        19.993134
Corolla      19.972057
Altima       19.963760
Name: proportion, dtype: float64
Car_Make
Honda        20.036812
Ford         20.034237
Chevrolet    20.024414
Toyota       19.961853
Nissan       19.942684
Name: proportion, dtype: float64
```

In [20]:

```
# Get value counts for car makes
car_make_counts = df['Car_Make'].value_counts()

# Create a pie chart
fig, ax = plt.subplots(figsize=(8, 8))
ax.pie(car_make_counts, labels=car_make_counts.index, autopct='%1.1f%%', startangle=90)
ax.axis('equal')
ax.set_title('Distribution of Car Makes')

plt.show()
```

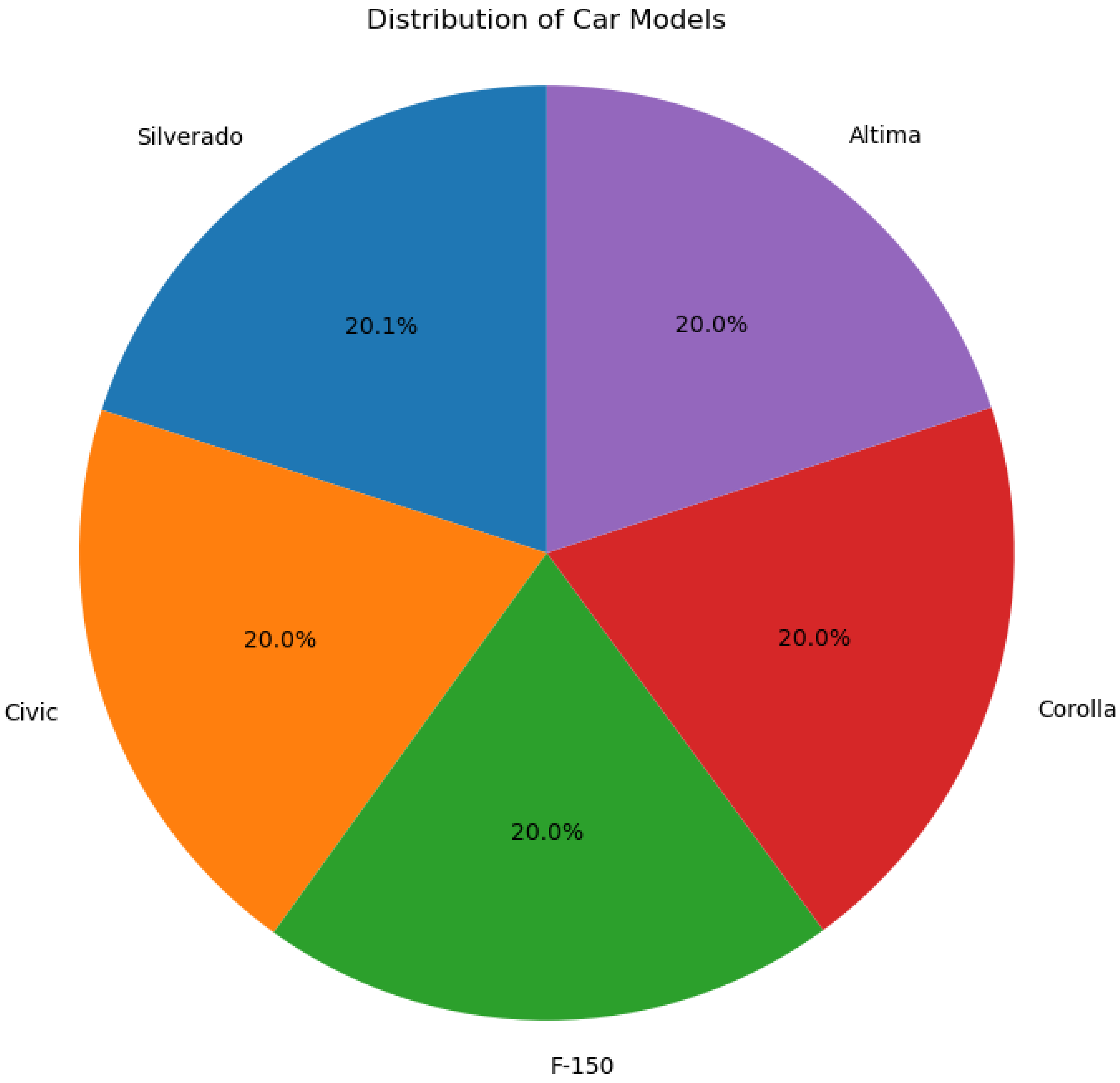


In [21]:

```
# Get value counts for car makes
car_make_counts = df['Car_Model'].value_counts()

# Create a pie chart
fig, ax = plt.subplots(figsize=(8, 8))
ax.pie(car_make_counts, labels=car_make_counts.index, autopct='%1.1f%%', startangle=90)
ax.axis('equal')
ax.set_title('Distribution of Car Models')

plt.show()
```



The 5 top car Makes in terms of market presence are Honda, Chevrolet, Toyota, Ford, and Nissan. Their market shares are distributed as follows: Honda at 20.03%, Chevrolet at 20.02%, Toyota at 20.01%, Ford at 19.99%, and Nissan at 19.96%.

Top Sales persons

In [23]:

```
top_salespersons = df['Sales_person'].value_counts().head(5)
print(top_salespersons)
```

```
Sales_person
Michael Smith        540
Michael Johnson      410
James Smith          347
David Smith          340
Christopher Smith    325
Name: count, dtype: int64
```

Most sold car model

In [24]:

```
most_sold_car_make = df['Car_Make'].value_counts().idxmax()
```

```
print("The most sold car make is:", most_sold_car_make)
```

```
The most sold car make is: Honda
```

In [25]:

```
most_sold_model = df['Car_Model'].value_counts().index[0]
print("The most sold car model is:", most_sold_model)
```

```
The most sold car model is: Silverado
```

Yearly Sales

In [26]:

```
yearly_sales = df.groupby('Car_Year')['Sale_Price'].sum().reset_index()
yearly_sales = yearly_sales.sort_values(by='Sale_Price', ascending=False)
print(yearly_sales)
```

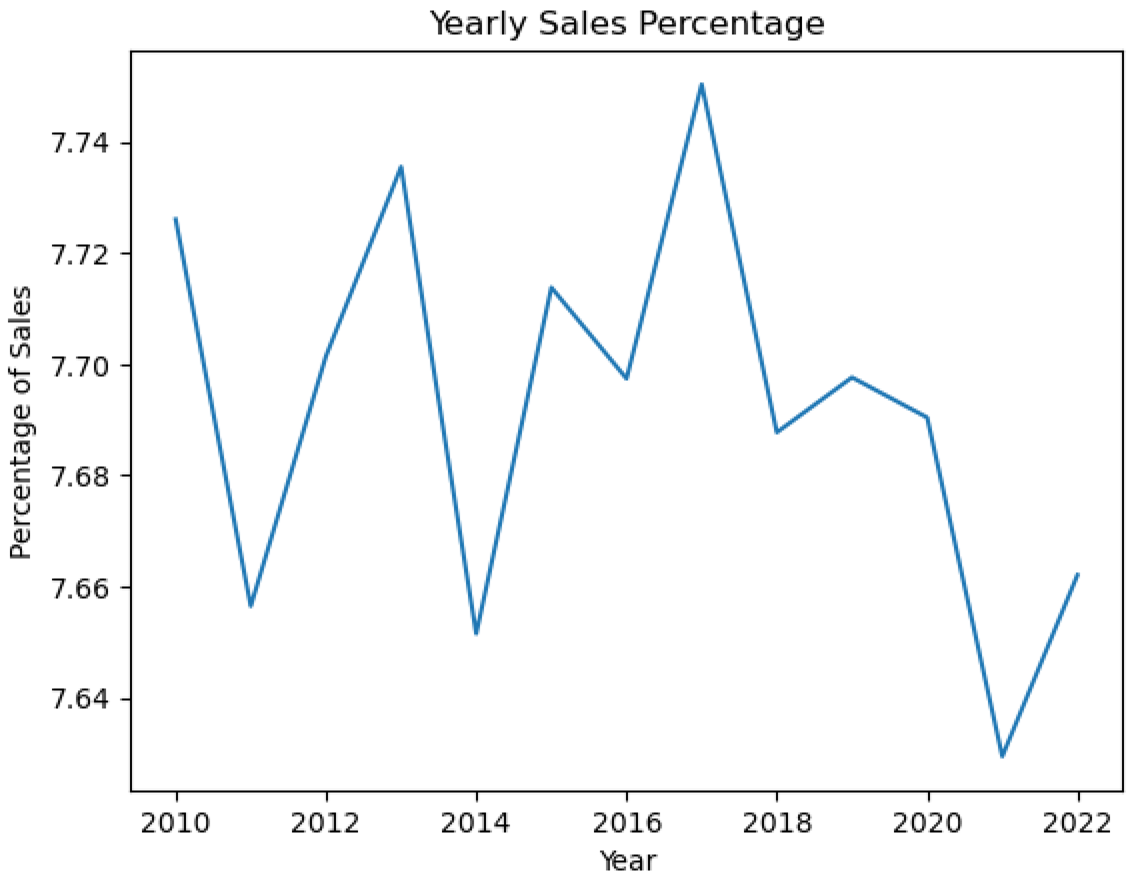
```
    Car_Year  Sale_Price
7       2017  2439544778
3       2013  2434875649
0       2010  2431900289
5       2015  2428038847
2       2012  2424144771
9       2019  2422945771
6       2016  2422877745
10      2020  2420677424
8       2018  2419835114
12      2022  2411756970
1       2011  2409996542
4       2014  2408436490
11      2021  2401486049
```

In [27]:

```
# Group the data by year and calculate the total sales for each year
yearly_sales = df.groupby('Car_Year')['Sale_Price'].sum()

# Calculate the percentage of sales for each year
yearly_sales_perc = yearly_sales / yearly_sales.sum() * 100

# Create a line chart to visualize the yearly sales percentage
plt.plot(yearly_sales_perc.index, yearly_sales_perc.values)
plt.xlabel('Year')
plt.ylabel('Percentage of Sales')
plt.title('Yearly Sales Percentage')
plt.show()
```



In 2015, the company achieved its peak sales performance, marking the highest annual sales figure. Conversely, the year 2022 recorded the lowest sales within the specified timeframe.

In [29]:
```
model_commission = df.groupby('Car_Model')['Commission_Earned'].mean()
print(model_commission)
```

```
Car_Model
Altima       2999.612387
Civic        2998.502515
Corolla      3006.177085
F-150        2998.251997
Silverado    3001.578450
Name: Commission_Earned, dtype: float64
```

In [31]:
```
salesperson_sales = df.groupby('Sales_person')['Sale_Price'].sum()
salesperson_commission = df.groupby('Sales_person')['Commission_Earned'].sum()

# Sort by total sales
salesperson_sales = salesperson_sales.sort_values(ascending=False)

# Sort by commission earned
salesperson_commission = salesperson_commission.sort_values(ascending=False)
```

In [32]:
```
salesperson_commission
```

Out[32]:
```
Sales_person
Michael Smith       1571856.41
Michael Johnson     1248024.79
James Smith         1073272.04
David Smith         1012556.25
Michael Williams     989580.64
                        ...
Juan Francis PhD        510.46
Molly Weaver            508.78
Felicia Mcneil          506.50
```

```
James Rogers DDS          503.33
Katie Stanley             502.86
Name: Commission_Earned, Length: 338696, dtype: float64
```

In [33]:

```
salesperson_total_commission = df.groupby('Sales_person')['Commission_Earned'].sum().sor
print(salesperson_total_commission)
```

```
Sales_person
Michael Smith         1571856.41
Michael Johnson       1248024.79
James Smith           1073272.04
David Smith           1012556.25
Michael Williams       989580.64
                         ...
Juan Francis PhD          510.46
Molly Weaver              508.78
Felicia Mcneil            506.50
James Rogers DDS          503.33
Katie Stanley             502.86
Name: Commission_Earned, Length: 338696, dtype: float64
```

After analyzing the car sales data, several insights were obtained. The dataset contained information about car sales, including the car model, car make, salesperson, sale price, commission rate, commission earned, and car year.

Some of the key findings from the analysis include:

The top 5 car models sold were the Silverado, Civic, Corolla, F-150, and Altima. The top 5 car makes sold were Honda, Chevrolet, Toyota, Ford, and Nissan. The top 5 salespersons in terms of commission earned were Michael Smith, Michael Johnson, David Smith, James Smith, and Robert Smith. The car model with the highest sales was the Silverado, followed closely by the Civic and Corolla. The car make with the highest sales was Toyota, followed closely by Honda and Chevrolet. There was a weak negative correlation between car year and sale price, suggesting that newer cars did not necessarily have higher sale prices. There was no strong correlation between car make and sale price. There was a slight difference in commission earned between different car models, with Corolla having the highest commission earned and F-150 having the lowest. Overall, this analysis provides some useful insights into the car sales data. However, further analysis could be performed to gain a deeper understanding of the data, such as analyzing the relationship between commission rate and commission earned, or examining the performance of individual salespersons over time.

In [ ]: