Import Packages

In [1]:
```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
import seaborn as sb
```

Import Dataset

In [2]:
```
df=pd.read_csv(r"C:\Users\LENOVO\Downloads\sales\car_purchasing.csv",encoding=
df
```

Out[2]:

	customer name	customer e-mail	country	gender	age
0	Martina Avila	cubilia.Curae.Phasellus@quisaccumsanconvallis.edu	Bulgaria	0	41.851720
1	Harlan Barnes	eu.dolor@diam.co.uk	Belize	0	40.870623
2	Naomi Rodriquez	vulputate.mauris.sagittis@ametconsectetueradip...	Algeria	1	43.152897
3	Jade Cunningham	malesuada@dignissim.com	Cook Islands	1	58.271369
4	Cedric Leach	felis.ullamcorper.viverra@egetmollislectus.net	Brazil	1	57.313749
...	...	...	...	...	...
495	Walter	ligula@Cumsociis.ca	Nepal	0	41.462515
496	Vanna	Cum.sociis.natoque@Sedmolestie.edu	Zimbabwe	1	37.642000
497	Pearl	penatibus.et@massanonante.com	Philippines	1	53.943497
498	Nell	Quisque.varius@arcuVivamussit.net	Botswana	1	59.160509
499	Marla	Camaron.marla@hotmail.com	marlal	1	46.731152

500 rows × 9 columns

Preprocessing

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 500 entries, 0 to 499
Data columns (total 9 columns):
 #   Column               Non-Null Count  Dtype  
---  ------               --------------  -----  
 0   customer name        500 non-null    object 
 1   customer e-mail      500 non-null    object 
 2   country              500 non-null    object 
 3   gender               500 non-null    int64  
 4   age                  500 non-null    float64
 5   annual Salary        500 non-null    float64
 6   credit card debt     500 non-null    float64
 7   net worth            500 non-null    float64
 8   car purchase amount  500 non-null    float64
dtypes: float64(5), int64(1), object(3)
memory usage: 35.3+ KB
```

In [4]:
```
df.shape
```

Out[4]:
```
(500, 9)
```

In [5]:
```
df.head()
```

Out[5]:

	customer name	customer e-mail	country	gender	age	
0	Martina Avila	cubilia.Curae.Phasellus@quisaccumsanconvallis.edu	Bulgaria	0	41.851720	6281
1	Harlan Barnes	eu.dolor@diam.co.uk	Belize	0	40.870623	6664
2	Naomi Rodriquez	vulputate.mauris.sagittis@ametconsectetueradip...	Algeria	1	43.152897	5379
3	Jade Cunningham	malesuada@dignissim.com	Cook Islands	1	58.271369	7937
4	Cedric Leach	felis.ullamcorper.viverra@egetmollislectus.net	Brazil	1	57.313749	5972

In [6]:
```
df.tail()
```

Out[6]:

	customer name	customer e-mail	country	gender	age	annual Salary	
495	Walter	ligula@Cumsociis.ca	Nepal	0	41.462515	71942.40291	
496	Vanna	Cum.sociis.natoque@Sedmolestie.edu	Zimbabwe	1	37.642000	56039.49793	1
497	Pearl	penatibus.et@massanonante.com	Philippines	1	53.943497	68888.77805	1
498	Nell	Quisque.varius@arcuVivamussit.net	Botswana	1	59.160509	49811.99062	1
499	Marla	Camaron.marla@hotmail.com	marlal	1	46.731152	61370.67766	

In [7]:
```
df.isna().any()
```

Out[7]:
```
customer name          False
customer e-mail        False
country                False
gender                 False
age                    False
annual Salary          False
credit card debt       False
net worth              False
car purchase amount    False
dtype: bool
```

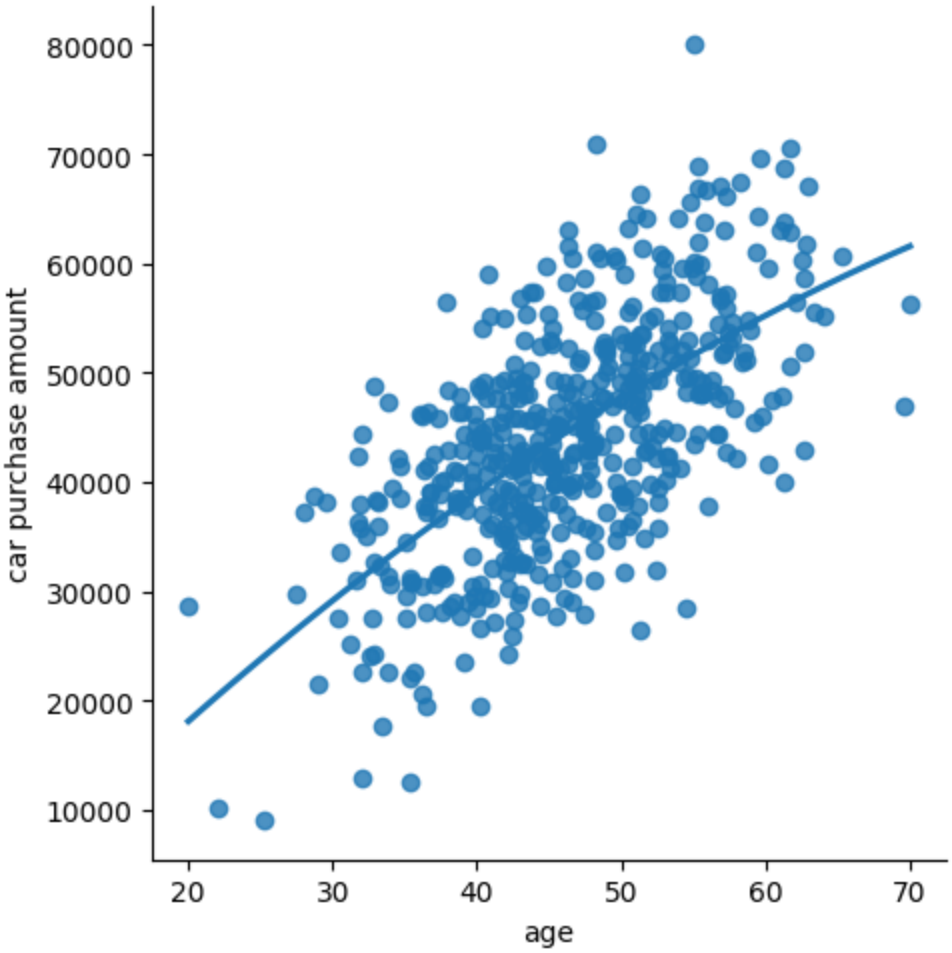
In [8]:
```
df.describe()
```

Out[8]:

	gender	age	annual Salary	credit card debt	net worth	car purchase amount
count	500.000000	500.000000	500.000000	500.000000	500.000000	500.000000
mean	0.506000	46.241674	62127.239608	9607.645049	431475.713625	44209.799218
std	0.500465	7.978862	11703.378228	3489.187973	173536.756340	10773.178744
min	0.000000	20.000000	20000.000000	100.000000	20000.000000	9000.000000
25%	0.000000	40.949969	54391.977195	7397.515792	299824.195900	37629.896040
50%	1.000000	46.049901	62915.497035	9655.035568	426750.120650	43997.783390
75%	1.000000	51.612263	70117.862005	11798.867487	557324.478725	51254.709517
max	1.000000	70.000000	100000.000000	20000.000000	1000000.000000	80000.000000

In [9]:
```
sb.lmplot(x="age",y="car purchase amount",data=df,order=2,ci=None)
```

Out[9]:
```
<seaborn.axisgrid.FacetGrid at 0x1ff91bcd290>
```



Model Building

In [10]:
```
x=df[["gender","age","annual Salary","net worth"]].values
y=df[["car purchase amount"]].values
```

In [11]:
```
from sklearn.linear_model import LinearRegression
```

In [12]:
```
s=LinearRegression()
```

In [13]:
```
s.fit(x,y)
```

Out[13]:
```
LinearRegression()
```

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

In [14]:
```
s.score(x,y)
```

Out[14]:
```
0.9999999812135176
```