

```
In [3]: import numpy as np
import pandas as pd
```

```
cars = pd.read_csv('Bikes.csv') cars
```

```
In [19]: car = pd.read_csv('car.csv')
bike = pd.read_csv('Bikes.csv')
evbike = pd.read_csv('Electric bikes.csv')
evcar = pd.read_csv('ELECTRIC CARS.csv')
car.head()
```

Out [19]:

	MODEL	COMPANY	PRICE(LAKHS) EX-SHOWROOM	FUEL TYPE	MILAGE (Kmpl)	SAFTY(star)	TYPE	BOOTSPAC
0	Baleno	suzuki	6.14	petrol	24.0	2.0	hatchback	339.
1	Swift	suzuki	5.93	petrol	24.0	1.0	hatchback	268.
2	Vitara Brezza	suzuki	7.69	petrol	19.0	3.0	compact suv	328.
3	ertiga	suzuki	8.12	petrol	19.0	2.0	mpv	209.
4	ertiga	suzuki	9.87	cng	24.0	2.0	mpv	209.

```
In [12]: car['TYPE'].value_counts()
```

```
Out [12]: sedan          55
suv          41
compact suv    33
hatchback     28
mpv           25
sports        25
off road       6
luxury         5
limousine      2
suv sports     2
Business       1
sports sedan   1
hatchback sports 1
Name: TYPE, dtype: int64
```

```
In [21]: car = car[['MODEL', 'COMPANY', 'PRICE(LAKHS) EX-SHOWROOM', 'FUEL TYPE', 'MILAGE (Kmpl)', 'SAFTY(star)', 'TYPE', 'BOOTSPACE', 'ENGINE CC']]
```

Out [21]:

	MODEL	COMPANY	PRICE(LAKHS) EX-SHOWROOM	FUEL TYPE	MILAGE (Kmpl)	SAFTY(star)	TYPE	BOOT
0	Baleno	suzuki	6.14	petrol	24.0	2.0	hatchback	
1	Swift	suzuki	5.93	petrol	24.0	1.0	hatchback	
2	Vitara Brezza	suzuki	7.69	petrol	19.0	3.0	compact suv	
3	ertiga	suzuki	8.12	petrol	19.0	2.0	mpv	
4	ertiga	suzuki	9.87	cng	24.0	2.0	mpv	
...
220	3 DOOR	MINI Cooper	39.00	petrol	17.0	3.0	hatchback	
221	countryman	MINI Cooper	41.00	petrol	14.0	4.0	compact suv	
222	DBX	Aston Martin	382.00	petrol	NaN	2.0	sports	
223	DB11	Aston Martin	380.00	petrol	NaN	2.0	sports	
224	Vantage	Aston Martin	300.00	petrol	NaN	2.0	sports	

225 rows × 9 columns

```
In [24]: car.isnull().sum()
```

```
Out [24]: MODEL                0
COMPANY                0
PRICE(LAKHS) EX-SHOWROOM    0
FUEL TYPE              0
MILAGE (Kmpl)          0
SAFTY(star)           0
TYPE                  0
BOOTSPACE             0
ENGINE CC             0
dtype: int64
```

```
In [23]: car.dropna(inplace=True)
```

```
/opt/anaconda3/lib/python3.9/site-packages/pandas/util/_decorators.py:311: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame
```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
return func(*args, **kwargs)
```

```
In [25]: car.duplicated().sum()
```

```
Out[25]: 0
```

```
In [29]: car.iloc[0].TYPE
```

```
Out[29]: 'hatchback'
```

```
In [34]: car['COMPANY'] = car['COMPANY'].apply(lambda x:[x.replace(" ", "")])
car['TYPE'] = car['TYPE'].apply(lambda x:[x.replace(" ", "")])
car['MODEL'] = car['MODEL'].apply(lambda x:[x.replace(" ", "")])
```

```
/var/folders/m6/1wwylt1n7r76r56qf8wb_x_c0000gn/T/ipykernel_1345/3040861267.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
car['COMPANY'] = car['COMPANY'].apply(lambda x:[x.replace(" ", "")])
/var/folders/m6/1wwylt1n7r76r56qf8wb_x_c0000gn/T/ipykernel_1345/3040861267.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
car['TYPE'] = car['TYPE'].apply(lambda x:[x.replace(" ", "")])
/var/folders/m6/1wwylt1n7r76r56qf8wb_x_c0000gn/T/ipykernel_1345/3040861267.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
car['MODEL'] = car['MODEL'].apply(lambda x:[x.replace(" ", "")])
```

In [46]: car

Out[46]:

	MODEL	COMPANY	PRICE(LAKHS) EX-SHOWROOM	FUEL TYPE	MILAGE (Kmpl)	SAFTY(star)	TY
0	[Baleno]	[suzuki]	6.14	petrol	24.0	2.0	[hatchba
1	[Swift]	[suzuki]	5.93	petrol	24.0	1.0	[hatchba
2	[VitaraBrezza]	[suzuki]	7.69	petrol	19.0	3.0	[compacts
3	[ertiga]	[suzuki]	8.12	petrol	19.0	2.0	[m
4	[ertiga]	[suzuki]	9.87	cng	24.0	2.0	[m
...	
217	[aventador]	[lamborghini]	625.00	petrol	8.0	2.0	[spoi
218	[huracan]	[lamborghini]	321.00	petrol	7.0	2.0	[spoi
219	[convertible]	[MINICooper]	45.00	petrol	17.0	3.0	[hatchbackspoi
220	[3DOOR]	[MINICooper]	39.00	petrol	17.0	3.0	[hatchba
221	[countryman]	[MINICooper]	41.00	petrol	14.0	4.0	[compacts

215 rows × 9 columns

```
In [51]: car['MODEL'] = car['MODEL'].apply(lambda x: " ".join(x))
car['COMPANY'] = car['COMPANY'].apply(lambda x: " ".join(x))
car['TYPE'] = car['TYPE'].apply(lambda x: " ".join(x))
```

```
/var/folders/m6/1wwylt1n7r76r56qf8wb_x_c0000gn/T/ipykernel_1345/2560353549.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
car['MODEL'] = car['MODEL'].apply(lambda x: " ".join(x))
/var/folders/m6/1wwylt1n7r76r56qf8wb_x_c0000gn/T/ipykernel_1345/2560353549.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
car['COMPANY'] = car['COMPANY'].apply(lambda x: " ".join(x))
/var/folders/m6/1wwylt1n7r76r56qf8wb_x_c0000gn/T/ipykernel_1345/2560353549.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
car['TYPE'] = car['TYPE'].apply(lambda x: " ".join(x))
```

In [52]: car

Out [52]:

	MODEL	COMPANY	PRICE(LAKHS) EX-SHOWROOM	FUEL TYPE	MILAGE (Kmpl)	SAFTY(star)	TYPE
0	Baleno	suzuki	6.14	petrol	24.0	2.0	hatchback
1	Swift	suzuki	5.93	petrol	24.0	1.0	hatchback
2	VitaraBrezza	suzuki	7.69	petrol	19.0	3.0	compactsuv
3	ertiga	suzuki	8.12	petrol	19.0	2.0	mpv
4	ertiga	suzuki	9.87	cng	24.0	2.0	mpv
...
217	aventador	lamborghini	625.00	petrol	8.0	2.0	sports
218	huracan	lamborghini	321.00	petrol	7.0	2.0	sports
219	convertible	MINICooper	45.00	petrol	17.0	3.0	hatchbacksports
220	3DOOR	MINICooper	39.00	petrol	17.0	3.0	hatchback
221	countryman	MINICooper	41.00	petrol	14.0	4.0	compactsuv

215 rows × 9 columns

```
In [53]: car['MODEL'] = car['MODEL'].apply(lambda x:x.lower())
car['COMPANY'] = car['COMPANY'].apply(lambda x:x.lower())
car['TYPE'] = car['TYPE'].apply(lambda x:x.lower())
```

```
/var/folders/m6/1wwylt1n7r76r56qf8wb_x_c0000gn/T/ipykernel_1345/1701459985.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
car['MODEL'] = car['MODEL'].apply(lambda x:x.lower())
/var/folders/m6/1wwylt1n7r76r56qf8wb_x_c0000gn/T/ipykernel_1345/1701459985.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
car['COMPANY'] = car['COMPANY'].apply(lambda x:x.lower())
/var/folders/m6/1wwylt1n7r76r56qf8wb_x_c0000gn/T/ipykernel_1345/1701459985.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
car['TYPE'] = car['TYPE'].apply(lambda x:x.lower())
```


In [54]: `car`

Out [54]:

	MODEL	COMPANY	PRICE(LAKHS) EX-SHOWROOM	FUEL TYPE	MILAGE (Kmpl)	SAFTY(star)	TYPE
0	baleno	suzuki	6.14	petrol	24.0	2.0	hatchback
1	swift	suzuki	5.93	petrol	24.0	1.0	hatchback
2	vitara	suzuki	7.69	petrol	19.0	3.0	compact
3	ertiga	suzuki	8.12	petrol	19.0	2.0	mpv
4	ertiga	suzuki	9.87	cng	24.0	2.0	mpv
...
217	aventador	lamborghini	625.00	petrol	8.0	2.0	sports
218	huracan	lamborghini	321.00	petrol	7.0	2.0	sports
219	convertible	minicooper	45.00	petrol	17.0	3.0	hatchback
220	3door	minicooper	39.00	petrol	17.0	3.0	hatchback
221	countryman	minicooper	41.00	petrol	14.0	4.0	compact

215 rows × 9 columns

In [68]: `car['tag1']=car['COMPANY']+' '+car['PRICE(LAKHS) EX-SHOWROOM'].astype`

```
/var/folders/m6/1wwylt1n7r76r56qf8wb_x_c0000gn/T/ipykernel_1345/3168540506.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
car['tag1']=car['COMPANY']+' '+car['PRICE(LAKHS) EX-SHOWROOM'].a
stype(str)+' '+car['FUEL TYPE']+' '+car['MILAGE (Kmpl)'].astype(st
r)+' '+car['SAFTY(star)'].astype(str)+' '+car['TYPE']+' '+car['B00
TSPACE'].astype(str)+' '+car['ENGINE CC'].astype(str)
```

In [69]: car

Out [69]:

	MODEL	COMPANY	PRICE(LAKHS) EX-SHOWROOM	FUEL TYPE	MILAGE (Kmpl)	SAFTY(star)	TYPE
0	baleno	suzuki	6.14	petrol	24.0	2.0	hatchback
1	swift	suzuki	5.93	petrol	24.0	1.0	hatchback
2	vitarabrezza	suzuki	7.69	petrol	19.0	3.0	compactsuv
3	ertiga	suzuki	8.12	petrol	19.0	2.0	mpv
4	ertiga	suzuki	9.87	cng	24.0	2.0	mpv
...
217	aventador	lamborghini	625.00	petrol	8.0	2.0	sports
218	huracan	lamborghini	321.00	petrol	7.0	2.0	sports
219	convertible	minicooper	45.00	petrol	17.0	3.0	hatchbacksports
220	3door	minicooper	39.00	petrol	17.0	3.0	hatchback
221	countryman	minicooper	41.00	petrol	14.0	4.0	compactsuv

215 rows × 10 columns

In [71]: car['tag1'][0]

Out [71]: 'suzuki 6.14 petrol 24.0 2.0 hatchback 339.0 1197'

```
In [73]: new_car = car[['MODEL', 'tag1']]
new_car
```

Out [73]:

	MODEL	tag1
0	baleno	suzuki 6.14 petrol 24.0 2.0 hatchback 339.0 1197
1	swift	suzuki 5.93 petrol 24.0 1.0 hatchback 268.0 1197
2	vitara	suzuki 7.69 petrol 19.0 3.0 compact 328.0 1462
3	ertiga	suzuki 8.12 petrol 19.0 2.0 mpv 209.0 1462
4	ertiga	suzuki 9.87 cng 24.0 2.0 mpv 209.0 1462
...
217	aventador	lamborghini 625.0 petrol 8.0 2.0 sports 110.0 ...
218	huracan	lamborghini 321.0 petrol 7.0 2.0 sports 70.0 5204
219	convertible	minicooper 45.0 petrol 17.0 3.0 hatchback 211....
220	3door	minicooper 39.0 petrol 17.0 3.0 hatchback 211....
221	countryman	minicooper 41.0 petrol 14.0 4.0 compact 450...

215 rows × 2 columns

```
In [74]: from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer(max_features=5000, stop_words='english')
```

```
In [80]: vectors=cv.fit_transform(new_car['tag1']).toarray()
```

```
In [81]: cv.fit_transform(new_car['tag1']).toarray().shape
```

Out [81]: (215, 346)

```
In [84]: vectors
```

```
Out [84]: array([[0, 0, 0, ..., 0, 0, 0],
                 [0, 0, 0, ..., 0, 0, 0],
                 [0, 0, 0, ..., 0, 0, 0],
                 ...,
                 [0, 0, 0, ..., 0, 0, 0],
                 [0, 0, 0, ..., 0, 0, 0],
                 [0, 0, 0, ..., 0, 0, 0]])
```

```
In [85]: cv.get_feature_names()
```

```
'470',  
'472',  
'475',  
'48',  
'480',  
'482',  
'483',  
'484',  
'49',  
'492',  
'494',  
'4999',  
'50',  
'500',  
'505',  
'506',  
'507',  
'51',  
'510',  
'515'
```

```
In [86]: import nltk  
from nltk.stem.porter import PorterStemmer  
ps = PorterStemmer()
```

```
In [89]: def stem(text):  
    y=[]  
    for i in text.split():  
        y.append(ps.stem(i))  
    return " ".join(y)
```

```
In [90]: from sklearn.metrics.pairwise import cosine_similarity
```

```
In [94]: similarity = cosine_similarity(vectors)
similarity
```

```
Out[94]: array([[1.          , 0.71428571, 0.28571429, ..., 0.14285714, 0.285
71429,
          0.28571429],
          [0.71428571, 1.          , 0.28571429, ..., 0.14285714, 0.285
71429,
          0.14285714],
          [0.28571429, 0.28571429, 1.          , ..., 0.14285714, 0.142
85714,
          0.28571429],
          ...,
          [0.14285714, 0.14285714, 0.14285714, ..., 1.          , 0.571
42857,
          0.42857143],
          [0.28571429, 0.28571429, 0.14285714, ..., 0.57142857, 1.
,
          0.42857143],
          [0.28571429, 0.14285714, 0.28571429, ..., 0.42857143, 0.428
57143,
          1.          ]])
```

```
In [92]: cosine_similarity(vectors).shape
```

```
Out[92]: (215, 215)
```

```
In [100]: similarity[0]
```

```
Out[100]: array([1.          , 0.71428571, 0.28571429, 0.28571429, 0.28571429,
 0.57142857, 0.57142857, 0.42857143, 0.57142857, 0.42857143,
 0.40089186, 0.28571429, 0.28571429, 0.42857143, 0.57142857,
 0.28571429, 0.14285714, 0.13363062, 0.14285714, 0.14285714,
 0.28571429, 0.13363062, 0.14285714, 0.          , 0.13363062,
 0.          , 0.28571429, 0.          , 0.26726124, 0.          ,
 0.42857143, 0.14285714, 0.14285714, 0.          , 0.28571429,
 0.          , 0.14285714, 0.28571429, 0.14285714, 0.26726124,
 0.13363062, 0.42857143, 0.14285714, 0.28571429, 0.14285714,
 0.28571429, 0.13363062, 0.13363062, 0.          , 0.          ,
 0.13363062, 0.13363062, 0.13363062, 0.13363062, 0.13363062,
 0.13363062, 0.13363062, 0.13363062, 0.          , 0.13363062,
 0.11952286, 0.14285714, 0.13363062, 0.71428571, 0.14285714,
 0.28571429, 0.13363062, 0.13363062, 0.14285714, 0.28571429,
 0.28571429, 0.28571429, 0.13363062, 0.28571429, 0.14285714,
 0.13363062, 0.14285714, 0.          , 0.14285714, 0.          ,
 0.42857143, 0.14285714, 0.14285714, 0.14285714, 0.13363062,
 0.26726124, 0.13363062, 0.28571429, 0.          , 0.          ,
 0.          , 0.          , 0.          , 0.14285714, 0.28571429,
 0.15430335, 0.28571429, 0.28571429, 0.14285714, 0.26726124,
 0.          , 0.13363062, 0.          , 0.          , 0.28571429,
 0.          , 0.          , 0.          , 0.13363062, 0.14285714,
 0.13363062, 0.13363062, 0.15430335, 0.26726124, 0.13363062,
 0.26726124, 0.13363062, 0.15430335, 0.14285714, 0.          ,
 0.14285714, 0.          , 0.14285714, 0.          , 0.          ,
 0.          , 0.14285714, 0.14285714, 0.14285714, 0.14285714,
 0.14285714, 0.28571429, 0.28571429, 0.14285714, 0.15430335,
 0.13363062, 0.14285714, 0.15430335, 0.14285714, 0.14285714,
 0.14285714, 0.14285714, 0.          , 0.          , 0.14285714,
 0.          , 0.          , 0.          , 0.          , 0.          ,
 0.          , 0.14285714, 0.          , 0.13363062, 0.28571429,
 0.13363062, 0.13363062, 0.13363062, 0.14285714, 0.15430335,
 0.14285714, 0.15430335, 0.14285714, 0.14285714, 0.15430335,
 0.14285714, 0.13363062, 0.14285714, 0.14285714, 0.14285714,
 0.26726124, 0.13363062, 0.14285714, 0.28571429, 0.13363062,
 0.          , 0.13363062, 0.          , 0.13363062, 0.          ,
 0.13363062, 0.13363062, 0.14285714, 0.          , 0.          ,
 0.14285714, 0.          , 0.13363062, 0.12598816, 0.25197632,
 0.          , 0.          , 0.13363062, 0.14285714, 0.          ,
 0.14285714, 0.14285714, 0.15430335, 0.14285714, 0.15430335,
 0.15430335, 0.15430335, 0.15430335, 0.15430335, 0.15430335,
 0.14285714, 0.14285714, 0.14285714, 0.          , 0.15430335,
 0.15430335, 0.15430335, 0.14285714, 0.28571429, 0.28571429])
```

In [106]: similarity[1]

```
Out[106]: array([0.71428571, 1.          , 0.28571429, 0.28571429, 0.28571429,
0.57142857, 0.57142857, 0.42857143, 0.57142857, 0.42857143,
0.26726124, 0.28571429, 0.28571429, 0.42857143, 0.57142857,
0.28571429, 0.14285714, 0.13363062, 0.14285714, 0.14285714,
0.28571429, 0.13363062, 0.14285714, 0.          , 0.13363062,
0.          , 0.28571429, 0.          , 0.13363062, 0.          ,
0.42857143, 0.14285714, 0.14285714, 0.          , 0.28571429,
0.          , 0.14285714, 0.28571429, 0.14285714, 0.13363062,
0.          , 0.42857143, 0.14285714, 0.28571429, 0.14285714,
0.28571429, 0.13363062, 0.13363062, 0.          , 0.          ,
0.          , 0.          , 0.13363062, 0.13363062, 0.13363062,
0.13363062, 0.13363062, 0.13363062, 0.          , 0.13363062,
0.          , 0.14285714, 0.13363062, 0.57142857, 0.14285714,
0.28571429, 0.13363062, 0.13363062, 0.14285714, 0.28571429,
0.28571429, 0.28571429, 0.          , 0.28571429, 0.14285714,
0.          , 0.14285714, 0.          , 0.14285714, 0.          ,
0.42857143, 0.14285714, 0.14285714, 0.14285714, 0.13363062,
0.13363062, 0.          , 0.28571429, 0.          , 0.          ,
0.          , 0.          , 0.          , 0.14285714, 0.14285714,
0.15430335, 0.28571429, 0.28571429, 0.14285714, 0.13363062,
0.          , 0.13363062, 0.          , 0.          , 0.14285714,
0.          , 0.          , 0.          , 0.13363062, 0.14285714,
0.13363062, 0.13363062, 0.15430335, 0.13363062, 0.13363062,
0.13363062, 0.13363062, 0.15430335, 0.14285714, 0.          ,
0.          , 0.          , 0.          , 0.          , 0.          ,
0.          , 0.14285714, 0.14285714, 0.14285714, 0.14285714,
0.14285714, 0.14285714, 0.14285714, 0.14285714, 0.15430335,
0.13363062, 0.14285714, 0.15430335, 0.14285714, 0.14285714,
0.14285714, 0.14285714, 0.          , 0.          , 0.          ,
0.          , 0.          , 0.          , 0.          , 0.          ,
0.          , 0.          , 0.          , 0.          , 0.          ,
0.          , 0.          , 0.          , 0.13363062, 0.14285714,
0.13363062, 0.13363062, 0.13363062, 0.14285714, 0.15430335,
0.14285714, 0.15430335, 0.14285714, 0.14285714, 0.15430335,
0.14285714, 0.13363062, 0.14285714, 0.14285714, 0.14285714,
0.13363062, 0.13363062, 0.14285714, 0.14285714, 0.13363062,
0.          , 0.13363062, 0.          , 0.13363062, 0.          ,
0.13363062, 0.          , 0.14285714, 0.          , 0.          ,
0.14285714, 0.          , 0.13363062, 0.12598816, 0.12598816,
0.          , 0.          , 0.13363062, 0.14285714, 0.          ,
0.14285714, 0.14285714, 0.15430335, 0.14285714, 0.15430335,
0.15430335, 0.15430335, 0.15430335, 0.15430335, 0.15430335,
0.14285714, 0.14285714, 0.          , 0.          , 0.15430335,
0.15430335, 0.15430335, 0.14285714, 0.28571429, 0.14285714])
```

```
In [121]: def recommend(car):
            car_index = new_car[new_car['MODEL']==car].index[0]
            distances = similarity[car_index]
            car_list = sorted(list(enumerate(distances)), reverse=True, key=lambda i: i[1])

            for i in car_list:
                print(new_car.iloc[i[0]].MODEL)
                print(i[0])
```

```
In [126]: recommend('baleno')
```

```
swift
1
glanza
63
dzire
5
wagonr
6
celerio
8
```

```
In [124]: new_car['MODEL'][63]
```

```
Out[124]: 'glanza'
```

```
In [ ]:
```