

PYTHON PROJECT ON HARDWARE DATA SET

Importing libraries

```
In [43]: import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
import seaborn as sns
```

```
In [ ]:
```

Loading the dataset

In [44]:

```
Data=pd.read_csv(r'Hardware.csv',encoding='latin')
Data
```

Out[44]:

	CATEGORY_ID	CATEGORY_NAME	PRODUCT_ID	PRODUCT_NAME	DESCRIPTION	DE
0	4	Mother Board	190	Supermicro X10SDV-8C-TLN4F	CPU:Xeon D-1541,Form Factor:Mini ITX, RAM Slots...	CI
1	4	Mother Board	190	Supermicro X10SDV-8C-TLN4F	CPU:Xeon D-1541,Form Factor:Mini ITX, RAM Slots...	CI
2	4	Mother Board	190	Supermicro X10SDV-8C-TLN4F	CPU:Xeon D-1541,Form Factor:Mini ITX, RAM Slots...	CI
3	4	Mother Board	190	Supermicro X10SDV-8C-TLN4F	CPU:Xeon D-1541,Form Factor:Mini ITX, RAM Slots...	CI
4	4	Mother Board	190	Supermicro X10SDV-8C-TLN4F	CPU:Xeon D-1541,Form Factor:Mini ITX, RAM Slots...	CI
...
151	4	Mother Board	152	Gigabyte X299 AORUS Gaming 3	CPU:LGA2066,Form Factor:ATX, RAM Slots:8, Max RA...	CP
152	4	Mother Board	152	Gigabyte X299 AORUS Gaming 3	CPU:LGA2066,Form Factor:ATX, RAM Slots:8, Max RA...	CP
153	4	Mother Board	152	Gigabyte X299 AORUS Gaming 3	CPU:LGA2066,Form Factor:ATX, RAM Slots:8, Max RA...	CP
154	4	Mother Board	152	Gigabyte X299 AORUS Gaming 3	CPU:LGA2066,Form Factor:ATX, RAM Slots:8, Max RA...	CP
155	4	Mother Board	152	Gigabyte X299 AORUS Gaming 3	CPU:LGA2066,Form Factor:ATX, RAM Slots:8, Max RA...	CP

156 rows × 22 columns



INTRODUCTION OF THE PROJECT

TARGET COLUMN == PRODUCT_NAME AND DESCRIPTION.

- PRODUCT_NAME:- PRODUCT_NAME consists names of various Mother Board and there Category_id.
- DESCRIPTION:- DESCRIPTION holds different parameters and details of various Mother Board that are atmost necessary.

PROBLEM STATEMENT

The aim and objective of this project is to perform analysis of categorical and numerical column in the dataset and visualise the distribution of CPU, Motherboard, Videogames across different prices, Quantity, Countries, States and Cities

Exploratory Data Analysis and Data cleaning



In [23]: #visualizing top 10 rows
Data.head(10)

Out[23]:

	CATEGORY_ID	CATEGORY_NAME	PRODUCT_ID	PRODUCT_NAME	DESCRIPTION	DESCR
0	4	Mother Board	190	Supermicro X10SDV-8C-TLN4F	CPU:Xeon D-1541,Form Factor:Mini ITX, RAM Slots...	CPU:ITX
1	4	Mother Board	190	Supermicro X10SDV-8C-TLN4F	CPU:Xeon D-1541,Form Factor:Mini ITX, RAM Slots...	CPU:ITX
2	4	Mother Board	190	Supermicro X10SDV-8C-TLN4F	CPU:Xeon D-1541,Form Factor:Mini ITX, RAM Slots...	CPU:ITX
3	4	Mother Board	190	Supermicro X10SDV-8C-TLN4F	CPU:Xeon D-1541,Form Factor:Mini ITX, RAM Slots...	CPU:ITX
4	4	Mother Board	190	Supermicro X10SDV-8C-TLN4F	CPU:Xeon D-1541,Form Factor:Mini ITX, RAM Slots...	CPU:ITX
5	4	Mother Board	62	Intel DP35DPM	CPU:LGA775,Form Factor:ATX, RAM Slots:4, Max RAM...	CPU:ATX
6	4	Mother Board	62	Intel DP35DPM	CPU:LGA775,Form Factor:ATX, RAM Slots:4, Max RAM...	CPU:ATX
7	4	Mother Board	62	Intel DP35DPM	CPU:LGA775,Form Factor:ATX, RAM Slots:4, Max RAM...	CPU:ATX
8	4	Mother Board	62	Intel DP35DPM	CPU:LGA775,Form Factor:ATX, RAM Slots:4, Max RAM...	CPU:ATX
9	4	Mother Board	62	Intel DP35DPM	CPU:LGA775,Form Factor:ATX, RAM Slots:4, Max RAM...	CPU:ATX

10 rows × 22 columns



In [24]: `#visualizing bottom 10 rows`
`Data.tail(10)`

Out[24]:

	CATEGORY_ID	CATEGORY_NAME	PRODUCT_ID	PRODUCT_NAME	DESCRIPTION	DE
146	4	Mother Board	151	Supermicro X10SAT-O	CPU:LGA1150,Form Factor:ATX, RAM Slots:4,Max RA...	CP
147	4	Mother Board	151	Supermicro X10SAT-O	CPU:LGA1150,Form Factor:ATX, RAM Slots:4,Max RA...	CP
148	4	Mother Board	151	Supermicro X10SAT-O	CPU:LGA1150,Form Factor:ATX, RAM Slots:4,Max RA...	CP
149	4	Mother Board	151	Supermicro X10SAT-O	CPU:LGA1150,Form Factor:ATX, RAM Slots:4,Max RA...	CP
150	4	Mother Board	151	Supermicro X10SAT-O	CPU:LGA1150,Form Factor:ATX, RAM Slots:4,Max RA...	CP
151	4	Mother Board	152	Gigabyte X299 AORUS Gaming 3	CPU:LGA2066,Form Factor:ATX, RAM Slots:8,Max RA...	CP
152	4	Mother Board	152	Gigabyte X299 AORUS Gaming 3	CPU:LGA2066,Form Factor:ATX, RAM Slots:8,Max RA...	CP
153	4	Mother Board	152	Gigabyte X299 AORUS Gaming 3	CPU:LGA2066,Form Factor:ATX, RAM Slots:8,Max RA...	CP
154	4	Mother Board	152	Gigabyte X299 AORUS Gaming 3	CPU:LGA2066,Form Factor:ATX, RAM Slots:8,Max RA...	CP
155	4	Mother Board	152	Gigabyte X299 AORUS Gaming 3	CPU:LGA2066,Form Factor:ATX, RAM Slots:8,Max RA...	CP

10 rows × 22 columns



In []:

In [14]: `#shows total no.of rows and columns`
`Data.shape`

Out[14]: (156, 22)

In []:

```
In [17]: #shows all the column names
```

```
Data.columns
```

```
Out[17]: Index(['CATEGORY_ID', 'CATEGORY_NAME', 'PRODUCT_ID', 'PRODUCT_NAME',
       'DESCRIPTION', 'DESCRIPTION - Detail 1', 'DESCRIPTION - Detail 2',
       'DESCRIPTION - Detail 3', 'DESCRIPTION - Detail 4', 'STANDARD_COST',
       'LIST_PRICE', 'COUNTRY_ID', 'REGION_ID', 'LOCATION_ID', 'WAREHOUSE_I
D',
       'QUANTITY', 'WAREHOUSE_NAME', 'ADDRESS', 'POSTAL_CODE', 'CITY', 'STAT
E',
       'COUNTRY_NAME'],
      dtype='object')
```

```
In [ ]:
```

```
In [18]: Data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 156 entries, 0 to 155
Data columns (total 22 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   CATEGORY_ID      156 non-null    int64  
 1   CATEGORY_NAME    156 non-null    object  
 2   PRODUCT_ID       156 non-null    int64  
 3   PRODUCT_NAME     156 non-null    object  
 4   DESCRIPTION      156 non-null    object  
 5   DESCRIPTION - Detail 1 156 non-null    object  
 6   DESCRIPTION - Detail 2 156 non-null    object  
 7   DESCRIPTION - Detail 3 156 non-null    object  
 8   DESCRIPTION - Detail 4 156 non-null    object  
 9   STANDARD_COST    156 non-null    float64 
 10  LIST_PRICE       156 non-null    float64 
 11  COUNTRY_ID       156 non-null    object  
 12  REGION_ID        156 non-null    int64  
 13  LOCATION_ID      156 non-null    int64  
 14  WAREHOUSE_ID     156 non-null    int64  
 15  QUANTITY         156 non-null    int64  
 16  WAREHOUSE_NAME   156 non-null    object  
 17  ADDRESS          156 non-null    object  
 18  POSTAL_CODE      156 non-null    object  
 19  CITY              156 non-null    object  
 20  STATE             156 non-null    object  
 21  COUNTRY_NAME     156 non-null    object  
dtypes: float64(2), int64(6), object(14)
memory usage: 26.9+ KB
```

```
In [ ]:
```

In [19]: #checking Null values
pd.isnull(Data)

Out[19]:

	CATEGORY_ID	CATEGORY_NAME	PRODUCT_ID	PRODUCT_NAME	DESCRIPTION	DESCRIPTION - D
0	False	False	False	False	False	False
1	False	False	False	False	False	False
2	False	False	False	False	False	False
3	False	False	False	False	False	False
4	False	False	False	False	False	False
...
151	False	False	False	False	False	False
152	False	False	False	False	False	False
153	False	False	False	False	False	False
154	False	False	False	False	False	False
155	False	False	False	False	False	False

156 rows × 22 columns



In []:

In [20]: pd.isnull(Data).sum()

Out[20]:

CATEGORY_ID	0
CATEGORY_NAME	0
PRODUCT_ID	0
PRODUCT_NAME	0
DESCRIPTION	0
DESCRIPTION - Detail 1	0
DESCRIPTION - Detail 2	0
DESCRIPTION - Detail 3	0
DESCRIPTION - Detail 4	0
STANDARD_COST	0
LIST_PRICE	0
COUNTRY_ID	0
REGION_ID	0
LOCATION_ID	0
WAREHOUSE_ID	0
QUANTITY	0
WAREHOUSE_NAME	0
ADDRESS	0
POSTAL_CODE	0
CITY	0
STATE	0
COUNTRY_NAME	0
dtype: int64	

In []:

In [21]: *# By using the describe() method we can get the statistical summary of the Data*
Data.describe()

Out[21]:

	CATEGORY_ID	PRODUCT_ID	STANDARD_COST	LIST_PRICE	REGION_ID	LOCATION_ID
count	156.0	156.000000	156.000000	156.000000	156.000000	156.000000
mean	4.0	137.096154	329.552244	410.044038	2.570513	11.647436
std	0.0	70.167337	128.994529	165.432935	0.496597	4.553346
min	4.0	31.000000	207.080000	280.980000	2.000000	6.000000
25%	4.0	62.750000	231.580000	287.000000	2.000000	9.000000
50%	4.0	146.500000	262.400000	301.990000	3.000000	11.000000
75%	4.0	188.250000	415.000000	499.990000	3.000000	13.000000
max	4.0	288.000000	664.290000	948.990000	3.000000	23.000000



In []:

In [22]: Data.describe(include="object")

Out[22]:

	CATEGORY_NAME	PRODUCT_NAME	DESCRIPTION	DESCRIPTION - Detail 1	DESCRIPTION - Detail 2
count	156	156	156	156	156
unique	1	41	22	11	6
top	Mother Board	Supermicro X10SDV-8C-TLN4F	CPU:LGA1151, Form Factor:ATX, RAM Slots:4, Max RA...	CPU:LGA2011-3	Form Factor:ATX
freq	156	5	27	41	74



In []:

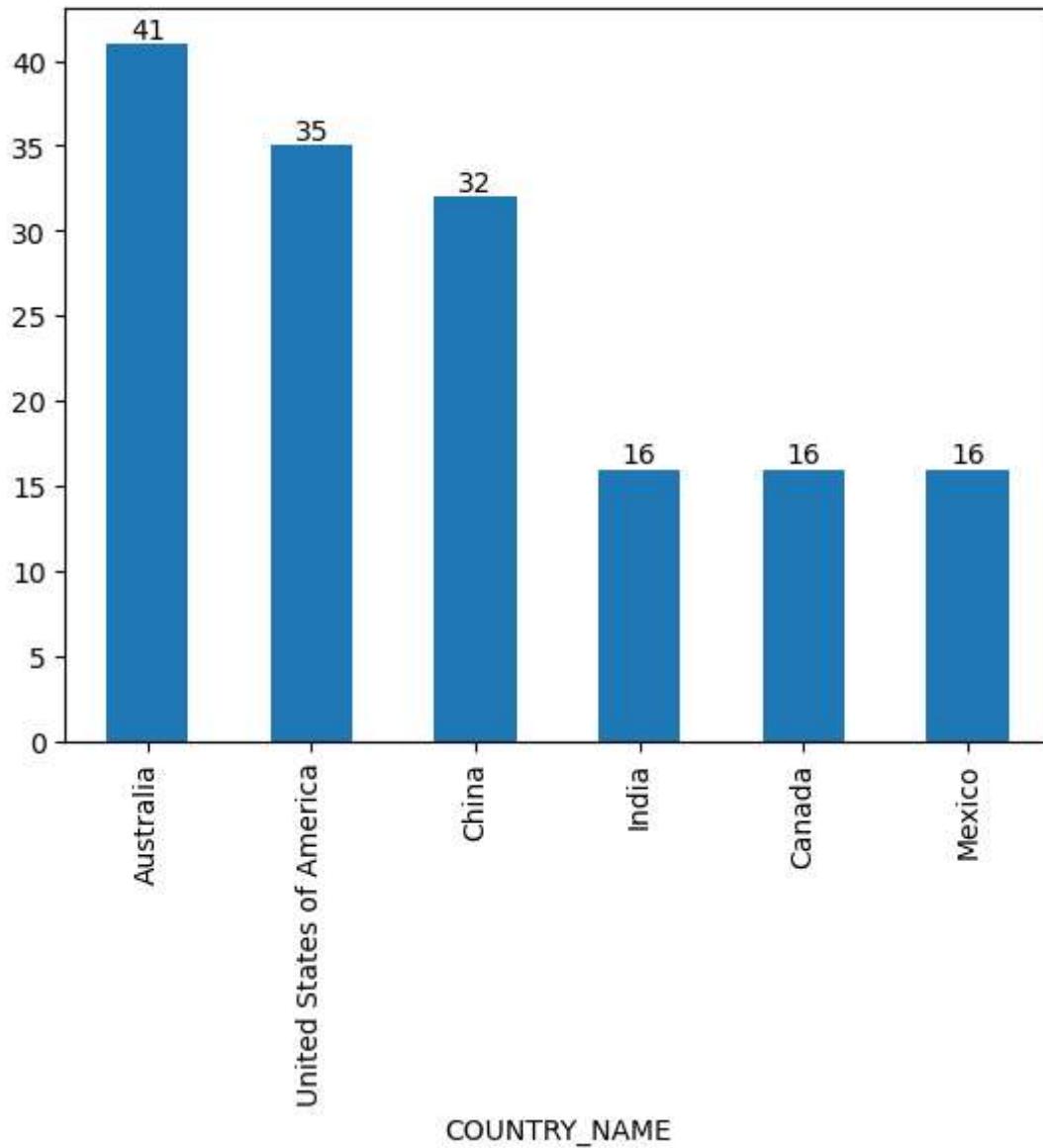
Data Analysis and Visualization

VAULE COUNT AS PER COUNTRY

```
In [27]: A = Data['COUNTRY_NAME'].value_counts().plot(kind='bar')

for i in A.containers:
    A.bar_label(i)

plt.show()
```

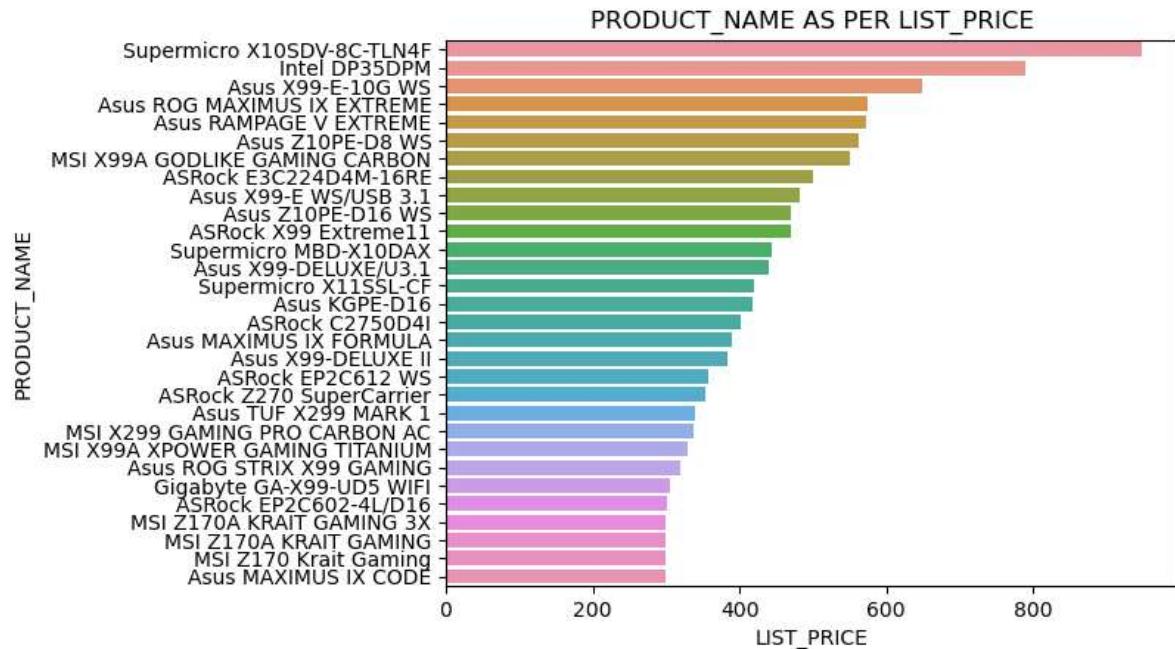


Observation:

- The above graph is showing us the total value count in the COUNTRY_NAME column.

1. PRODUCT AS PER LIST_PRICE

```
In [18]: A=Data.groupby(["PRODUCT_NAME"], as_index=False)[ 'LIST_PRICE'].mean().sort_values()
sns.barplot(data=A,y="PRODUCT_NAME",x="LIST_PRICE")
plt.title("PRODUCT_NAME AS PER LIST_PRICE")
plt.show()
```



Observation:-

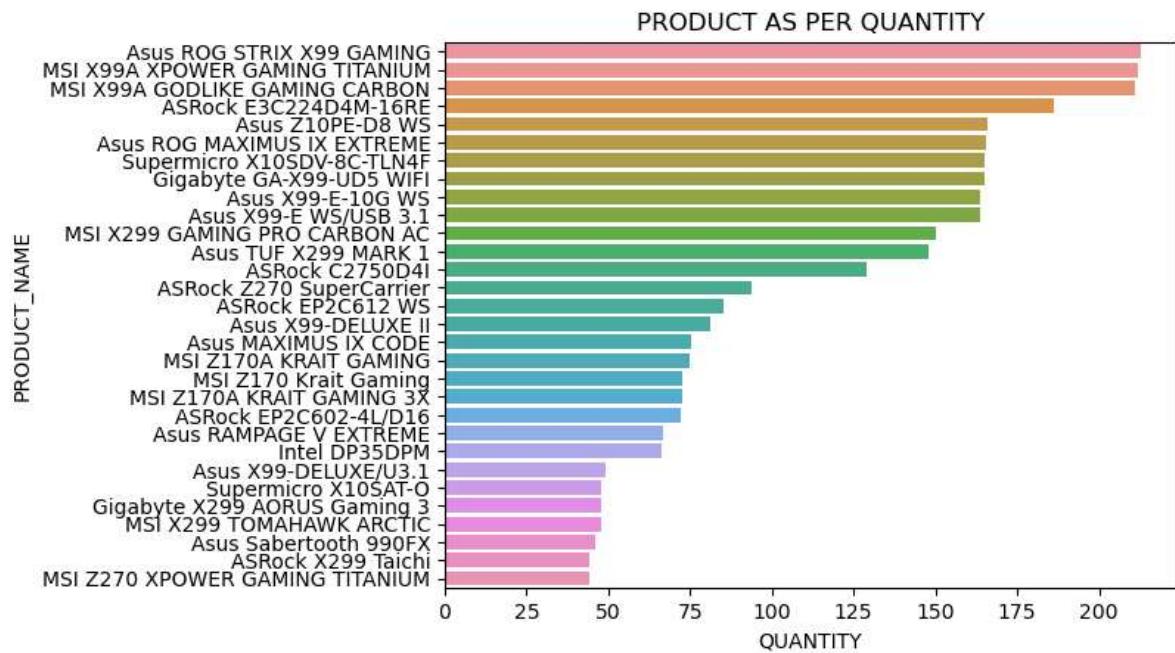
- In the above barplot graph we can see that Supermicro x10SDV-8C-TLN4F has the highest List_price and
- Intel DP35DPM being the second most List_price.
- Asus Z10PE-D16WS has the lowest List_price.

```
In [ ]:
```

2.PRODUCT AS PER QUANTITY

```
In [17]: A=Data.groupby(["PRODUCT_NAME"], as_index=False)[ 'QUANTITY' ].mean().sort_values
AX=sns.barplot(data=A,y="PRODUCT_NAME",x="QUANTITY")

plt.title("PRODUCT AS PER QUANTITY")
plt.show()
```

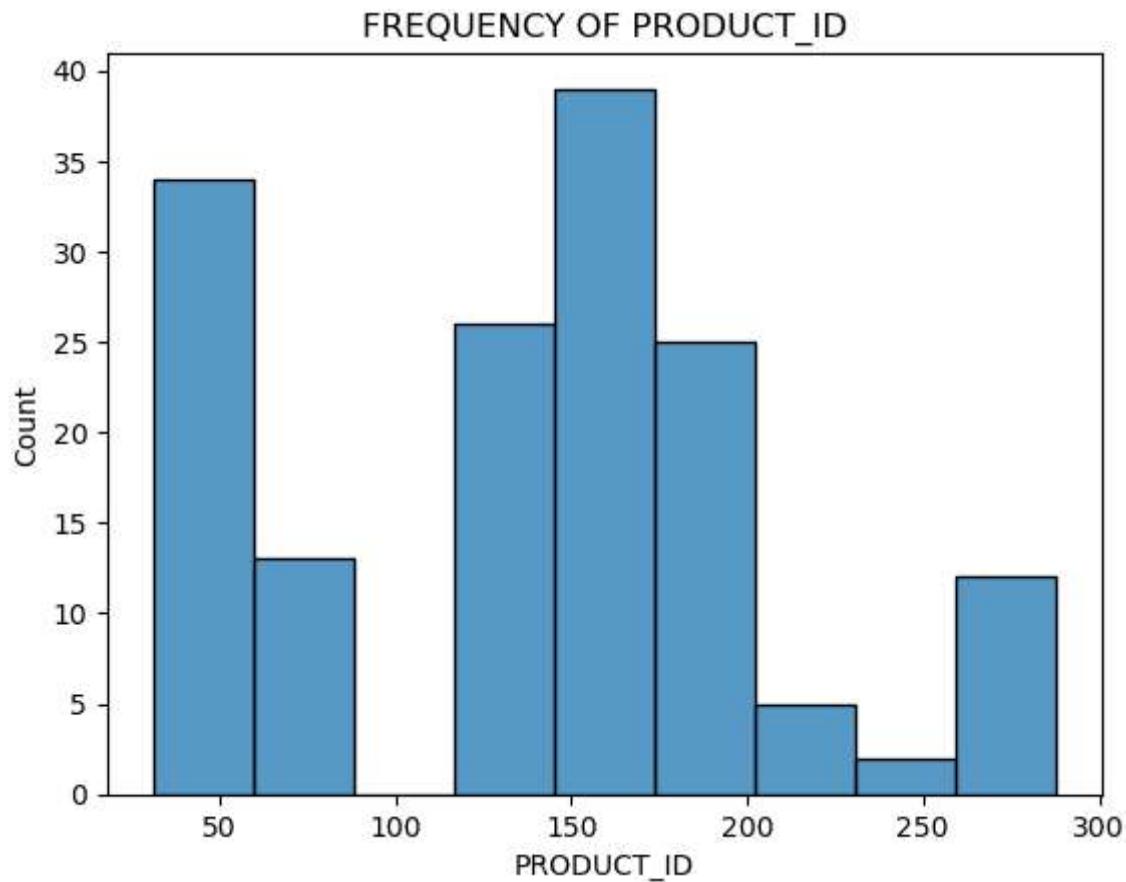


Observation:

- Asus ROG STRIX X99 GAMING, MSI X99A XPOWER GAMING TITANIUM, MSI X99A GODLIKE GAMING CARBON have the highest number of Quantity.
- In the last graph we saw that the list price of Supermicro x10SDV-8C-TLN4F was highest but in this graph we can see that the Quantity is above 150 so we can assume that the Quantity is average.

3.FREQUENCY FOR PRODUCT_ID

```
In [26]: sns.histplot(data=Data,x="PRODUCT_ID")
plt.title("FREQUENCY OF PRODUCT_ID")
plt.show()
```

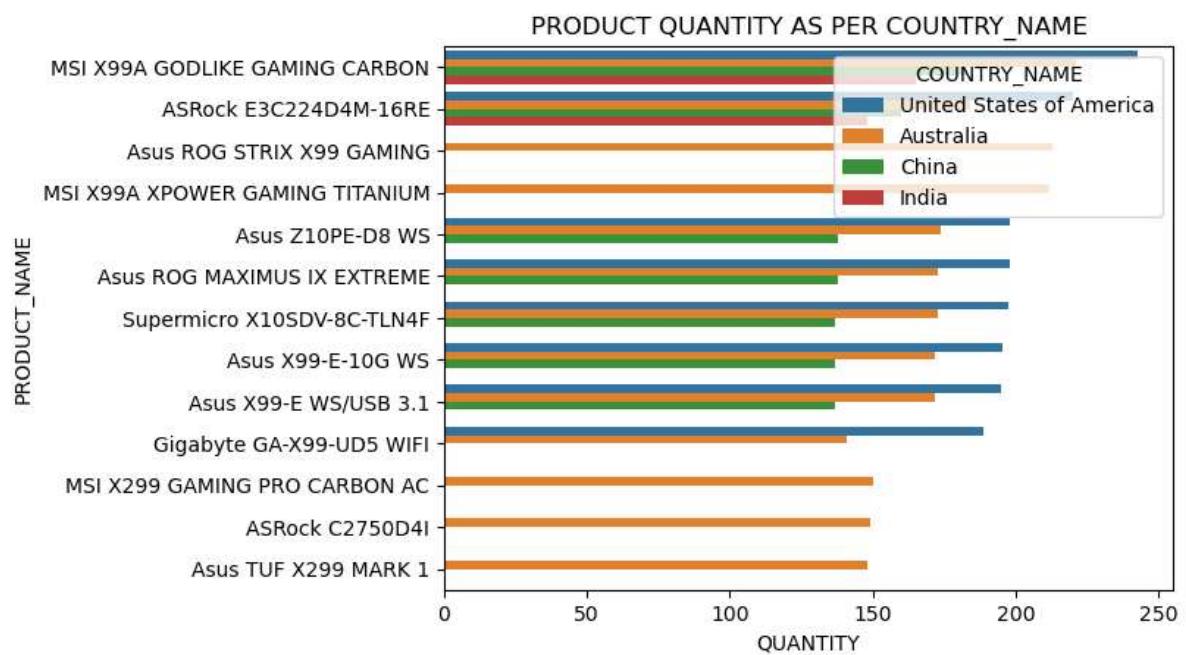


Observation:

- The following histplot graph is showing the frequency of the product_id.
- From this graph we can see that the highest frequency from the product_id is 150.
- We can also see that the frequency is starting from 120-200 where 150 has the highest frequency.

4. PRODUCT QUANTITY AS PER COUNTRY

```
In [16]: A=Data.groupby(["PRODUCT_NAME","COUNTRY_NAME"], as_index=False)[['QUANTITY']].mean()
AX=sns.barplot(data=A,y="PRODUCT_NAME",x="QUANTITY",hue="COUNTRY_NAME")
plt.title("PRODUCT QUANTITY AS PER COUNTRY_NAME")
plt.show()
```

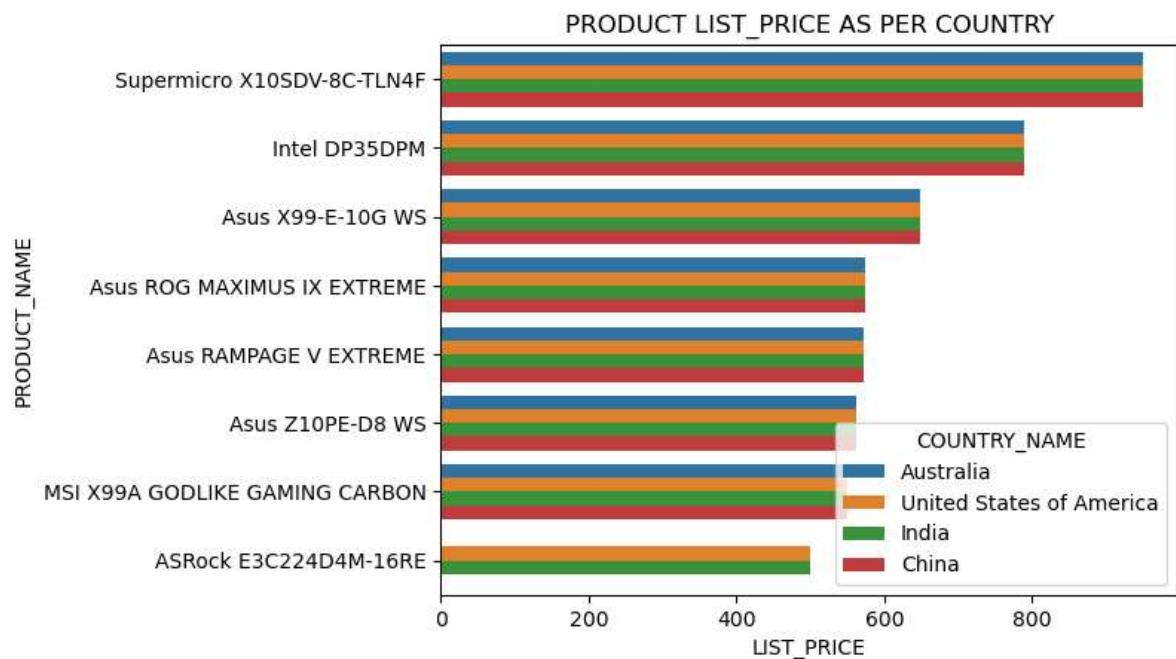


Observation:

- The above graph shows highest product as per Quantity in different Country.
- In the above graph we say that MSI X99A GODLIKE GAMING CARBON has the highest demand in United States of America.
- In the above graph we can that SUPERMICRO X10SDV-8C-TLN4F has highest price and its quantity is available at top7 product.

5. PRODUCT LIST_PRICE AS COUNTRY

```
In [19]: A=Data.groupby(["PRODUCT_NAME","COUNTRY_NAME"], as_index=False)[['LIST_PRICE']].  
AX=sns.barplot(data=A,y="PRODUCT_NAME",x="LIST_PRICE",hue="COUNTRY_NAME")  
plt.title("PRODUCT LIST_PRICE AS PER COUNTRY")  
plt.show()
```

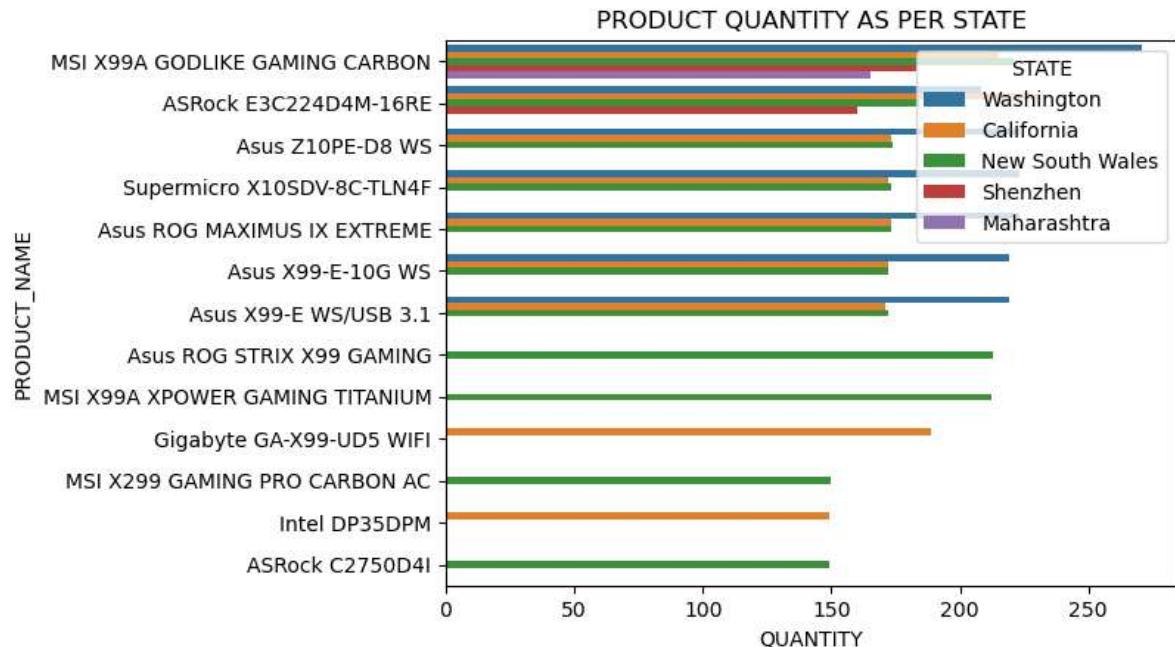


Observation:

- The above graph shows highest product as per List_price in different Country.
- In above graph we can see that SUPERMICRO X10SDV-8C-TLN4F has the highest price and is equally in demand in every Country.
- From the graphs before we have seen that SUPERMICRO X10SDV-8C-TLN4F is constantly in top 10 products.

6.PRODUCT QUANTITY AS PER STATE

```
In [11]: A=Data.groupby(["PRODUCT_NAME","STATE"], as_index=False)[['QUANTITY']].mean().so
AX=sns.barplot(data=A,y="PRODUCT_NAME",x="QUANTITY",hue="STATE")
plt.title("PRODUCT QUANTITY AS PER STATE")
plt.show()
```

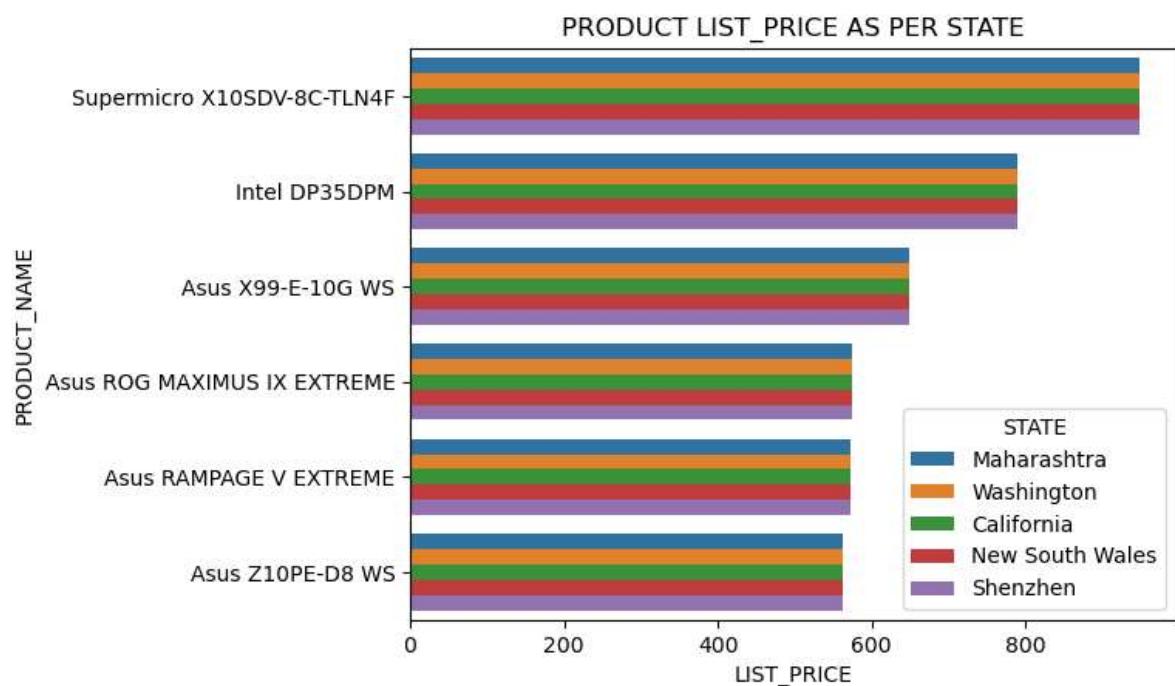


Observation:

- In the above barplot graph shows the product Quantity is shown as per state.
- From the above graph we can see that only MSI X99A GODLIKE GAMING CARBON is available in all the states.
- As per the graph we can also see that Washington has the highest number of Quantity of MSI X99A GODLIKE GAMING CARBON and Maharashtra being the lowest number of quantity.

7. PRODUCT LIST_PRICE AS PER STATE

```
In [14]: A=Data.groupby(["PRODUCT_NAME","STATE"], as_index=False)[['LIST_PRICE']].mean().  
AX=sns.barplot(data=A,y="PRODUCT_NAME",x="LIST_PRICE",hue="STATE")  
plt.title("PRODUCT LIST_PRICE AS PER STATE")  
plt.show()
```

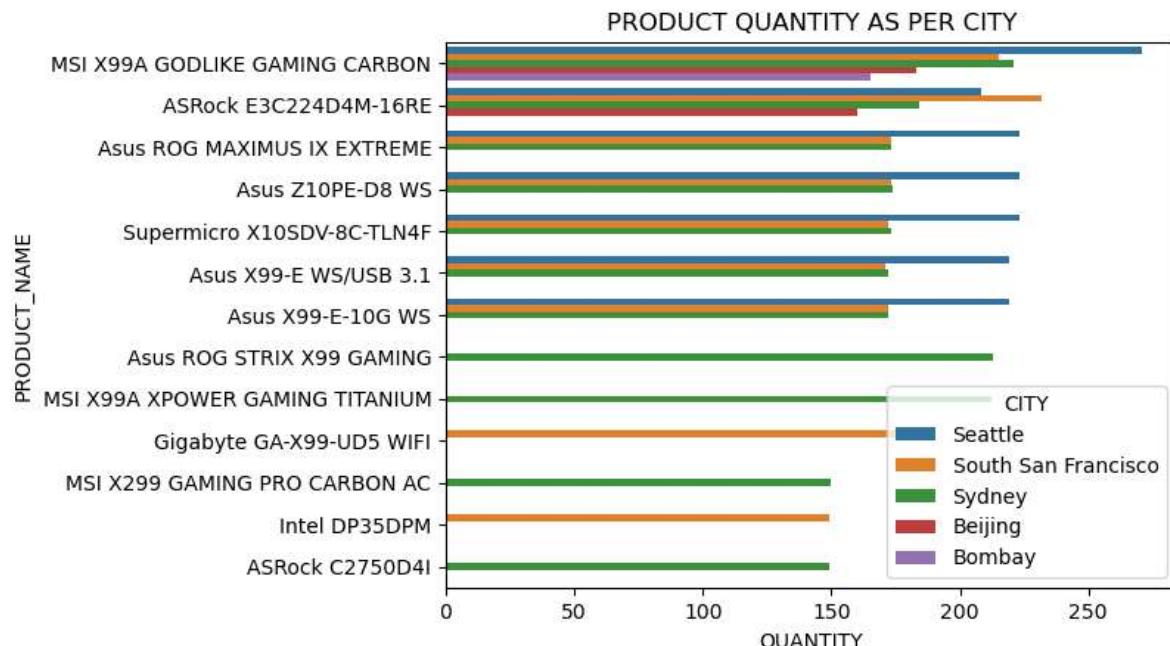


Observation:

- In the above barplot graph it shows the products List_price as per state
- The above graph also tells us that SUPERMICRO X10SDV-8C-TLN4F is equally priced in all the states.
- We can also see that the product SUPERMICRO X10SDV-8C-TLN4F is highly priced product in all the states.
- we can also analyze from the previous graph that SUPERMICRO X10SDV-8C-TLN4F is highly priced but does not have the Quantity in the country like India and state like Maharashtra

8. PRODUCT QUANTITY AS PER CITY

```
In [20]: A=Data.groupby(["PRODUCT_NAME", "CITY"], as_index=False)[['QUANTITY']].mean().sort_values('QUANTITY', ascending=False)
AX=sns.barplot(data=A,y="PRODUCT_NAME",x="QUANTITY",hue="CITY")
plt.title("PRODUCT QUANTITY AS PER CITY")
plt.show()
```

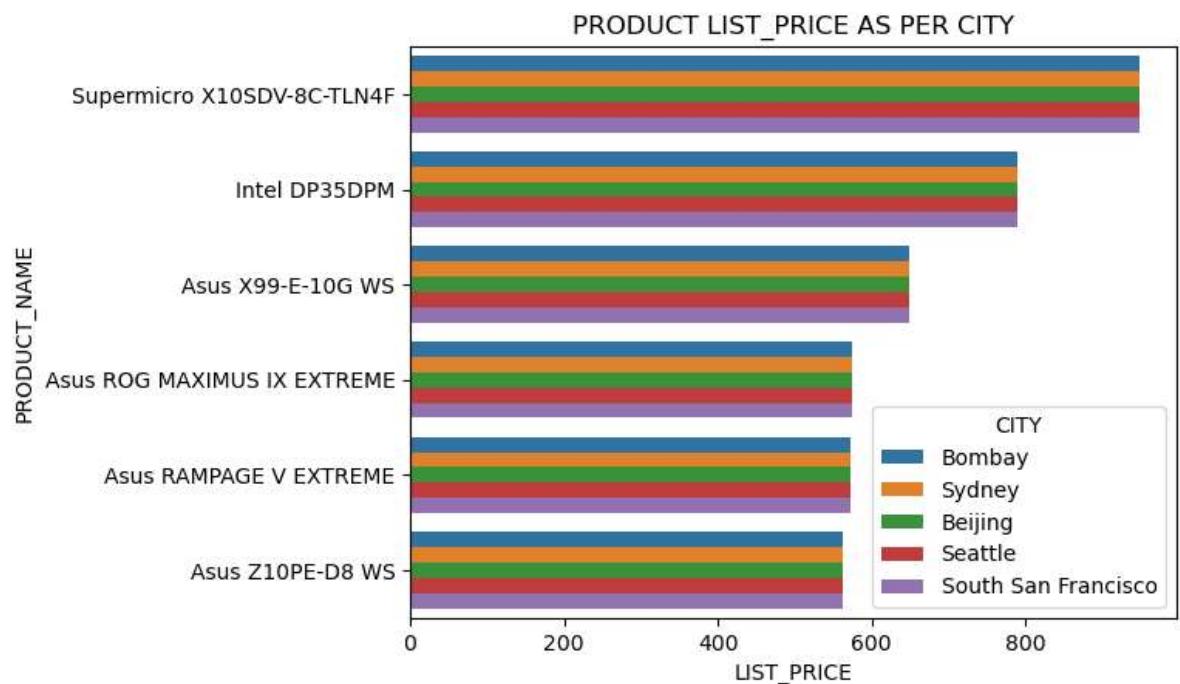


Observation:

- In the above barplot graph shows us the product quantity as per city.
- Even in this graph we can clearly analyze that MSI X99A GODLIKE GAMING CARBON has the highest quantity in the city of Seattle.
- We can also see that MSI X99A GODLIKE GAMING CARBON is available in every city where Seattle being the highest and Bombay to be the lowest

9. PRODUCT LIST_PRICE AS PER CITY

```
In [21]: A=Data.groupby(["PRODUCT_NAME", "CITY"], as_index=False)[['LIST_PRICE']].mean().sort_values('LIST_PRICE', ascending=False)
sns.barplot(data=A,y="PRODUCT_NAME",x="LIST_PRICE",hue="CITY")
plt.title("PRODUCT LIST_PRICE AS PER CITY")
plt.show()
```



Observation:

- In the above barplot graph it shows the products List_price as per city
- The above graph also tells us that SUPERMICRO X10SDV-8C-TLN4F is equally priced in all the city.
- We can also see that the product SUPERMICRO X10SDV-8C-TLN4F is highly priced product in all the city.

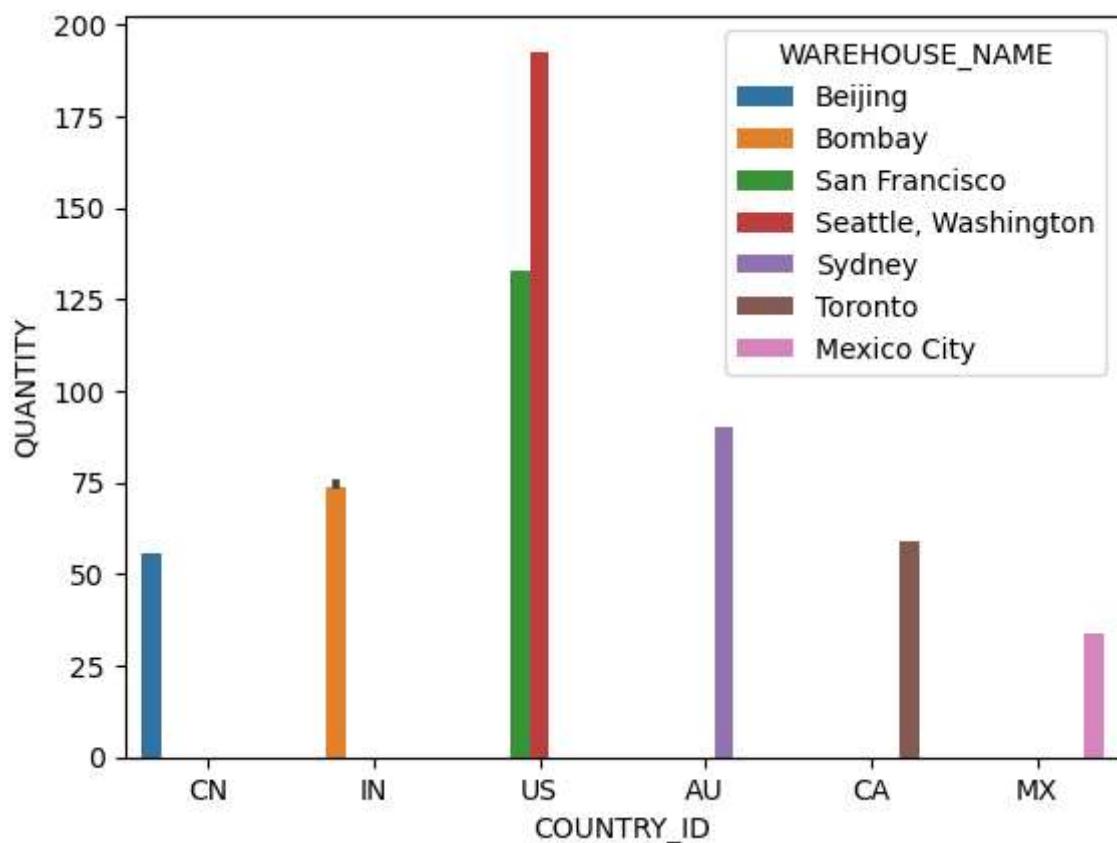
10. WAREHOUSE QUANTITY AS PER COUNTRY_ID

```
In [40]: sns.barplot(x="COUNTRY_ID", y="QUANTITY", data=Data,hue='WAREHOUSE_NAME',ci=True)
plt.show()
```

C:\Users\HP\AppData\Local\Temp\ipykernel_12148\3637149715.py:1: FutureWarning:

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

```
sns.barplot(x="COUNTRY_ID", y="QUANTITY", data=Data,hue='WAREHOUSE_NAME',ci=True) # hue parameter with numerical column
```

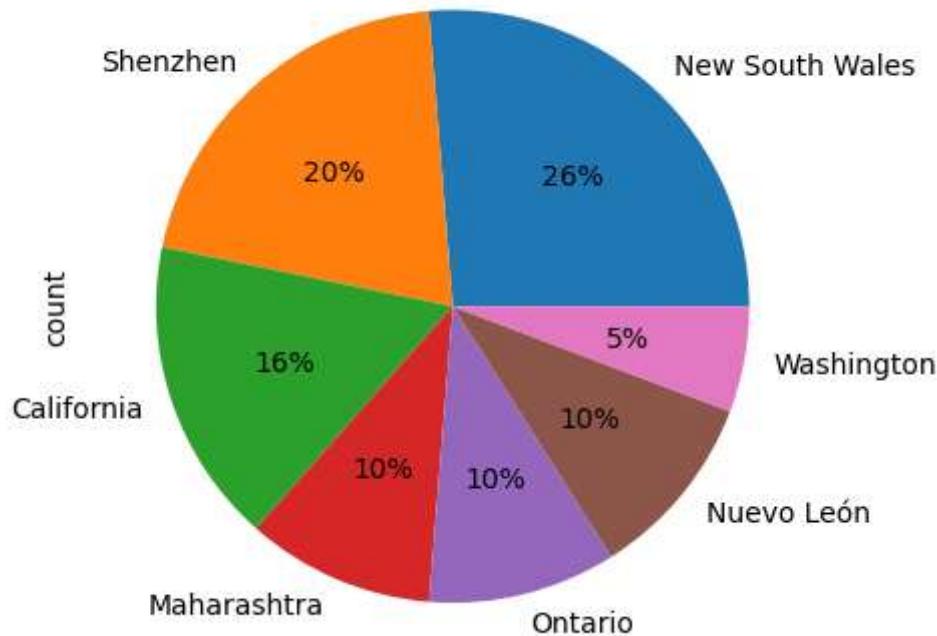


Observation:

- In this barplot graph we have shown that which country_id has the large amount of Quantity and where is there warehouse.
- As we can see that Country_id US has the biggest quantity and there warehouse is located in Seattle,Washington.

11. Value count as per State.

```
In [55]: Data['STATE'].value_counts().plot(kind='pie', autopct='%i%')  
plt.show()
```



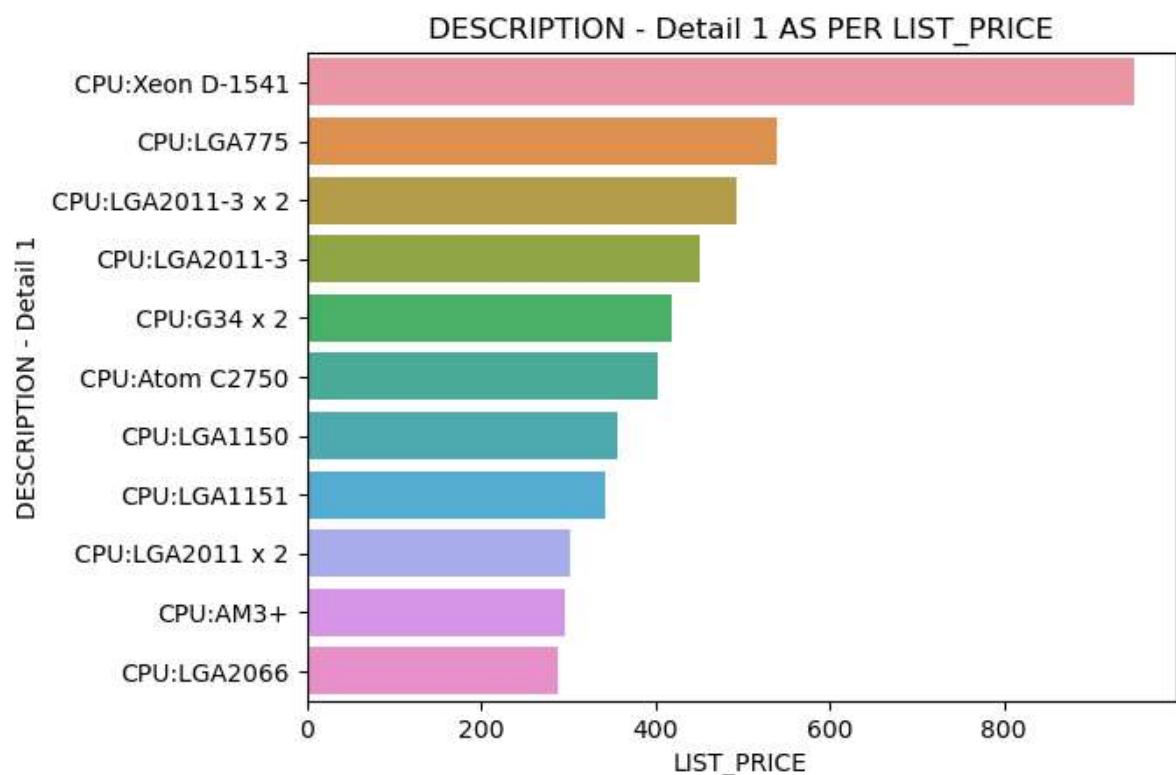
Observation:

- As per the pie chart it shows the percentage count of the buyers from its State
- As we can see that New South Wales has the largest percent of people.

DESCRIPTION

12.DESCRIPTION - DETAIL1

```
In [7]: A=Data.groupby(["DESCRIPTION - Detail 1"], as_index=False)[ 'LIST_PRICE'].mean()
sns.barplot(data=A,y="DESCRIPTION - Detail 1",x="LIST_PRICE")
plt.title("DESCRIPTION - Detail 1 AS PER LIST_PRICE")
plt.show()
```



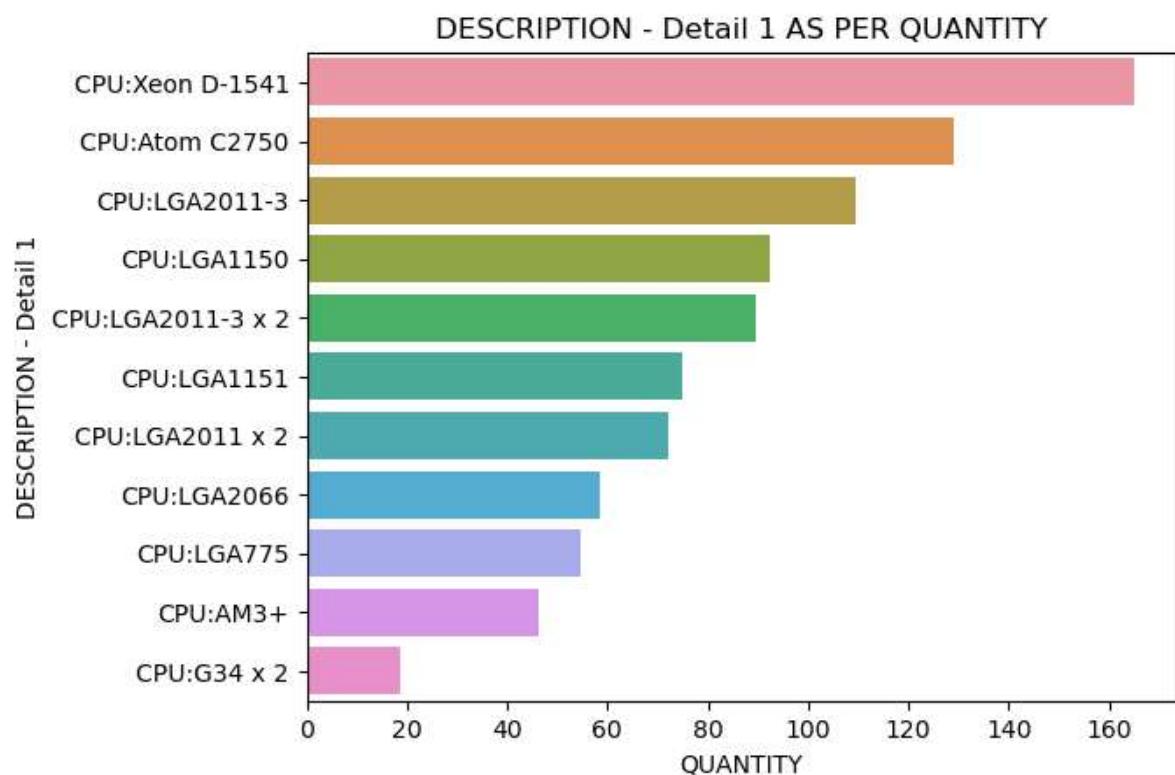
Observation

- In this above graph it is showing us the detailed description of all the CPU units as per List_price.
- From the graph shown we can analyze that CPU:XEON D-1541 is the most expensive CPU units.
- CPU:LGA2066 being the least expensive CPU unit.

```
In [ ]:
```

13.DESCRIPTION - Detail 1 AS PER QUANTITY

```
In [8]: A=Data.groupby(["DESCRIPTION - Detail 1"], as_index=False)[ 'QUANTITY'].mean()
sns.barplot(data=A,y="DESCRIPTION - Detail 1",x="QUANTITY")
plt.title("DESCRIPTION - Detail 1 AS PER QUANTITY")
plt.show()
```

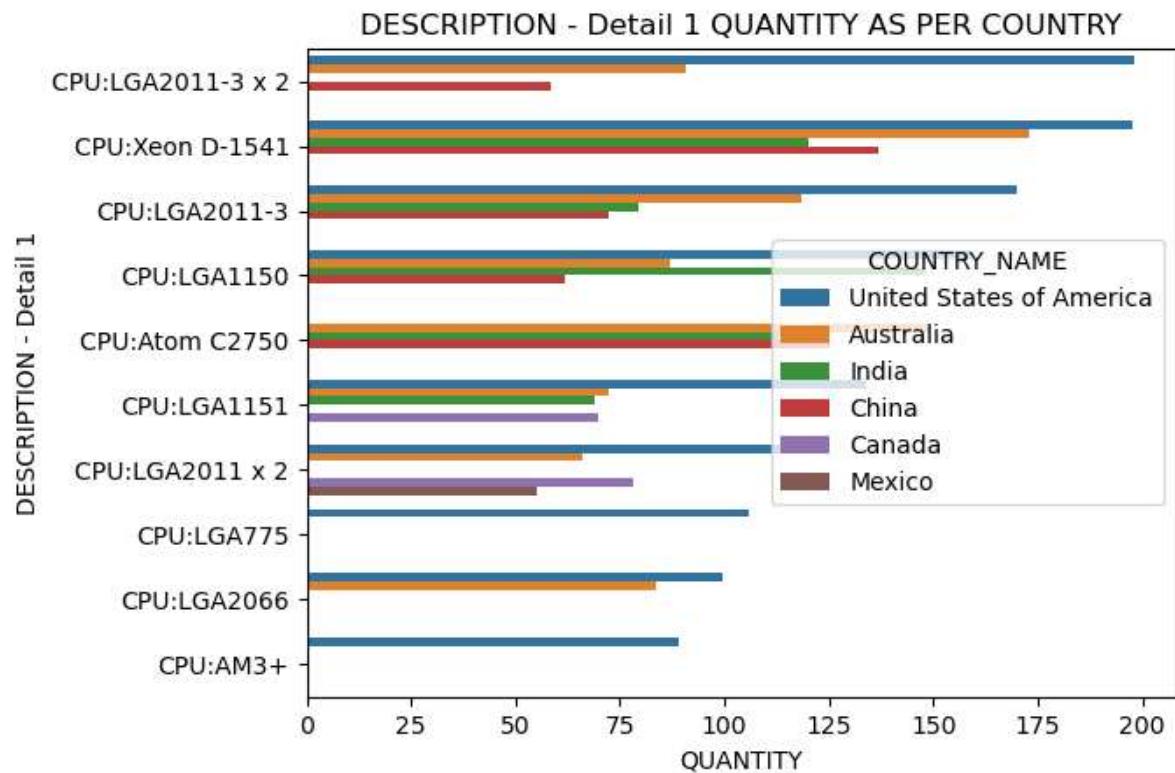


Observation:

- The above graph shows us the detailed description of the product's CPU units as per there Quantity.
- From the above graph we can analyze here that CPU:XEON D-1541 has the largest Quantity and CPU:G34x2 being the smallest Quantity from all.
- As per previous analysis we can see that CPU:XEON D-1541 is the most expensive and has the largest Quantity from all the other CPU units.

14.DESCRIPTION - Detail 1 QUANTITY AS PER COUNTRY

```
In [11]: A=Data.groupby(["DESCRIPTION - Detail 1","COUNTRY_NAME"], as_index=False)[['QUA
sns.barplot(data=A,y="DESCRIPTION - Detail 1",x="QUANTITY",hue="COUNTRY_NAME")
plt.title("DESCRIPTION - Detail 1 QUANTITY AS PER COUNTRY")
plt.show()
```

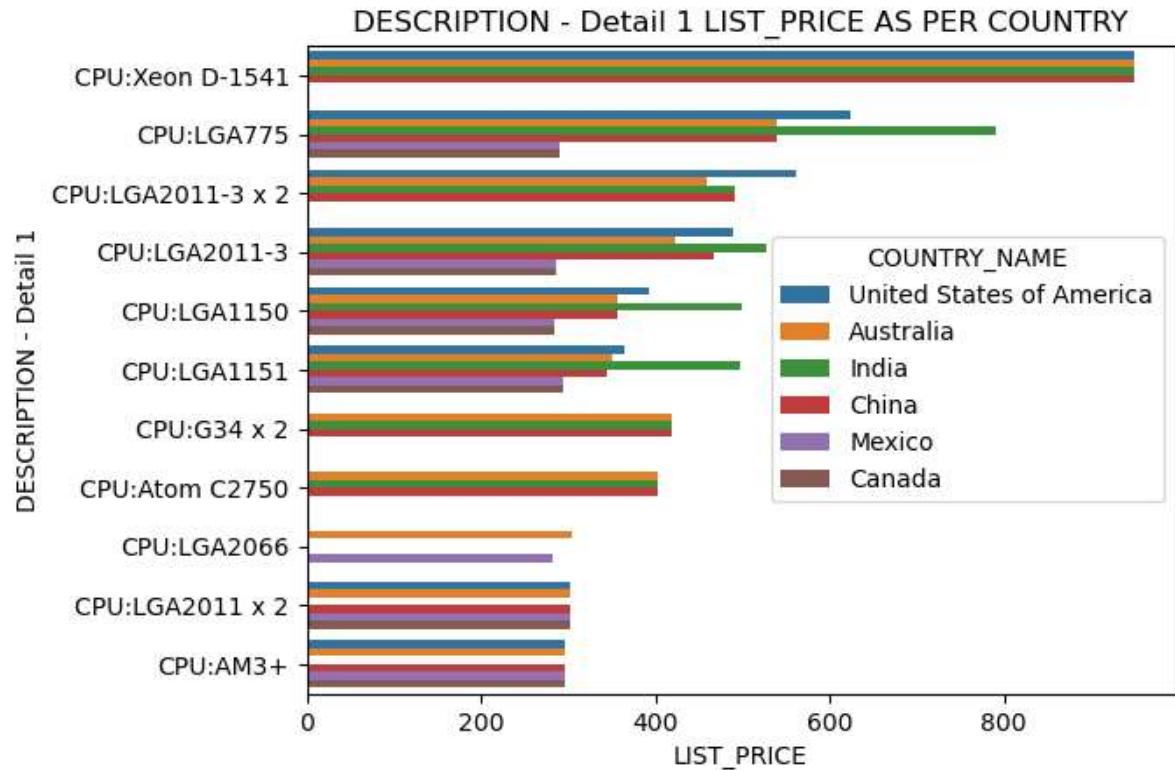


Observation:

- In this above graph it shows us the detailed description of products CPU units quantity as per country.
- In the above we can also analyze that CPU:LGA2011-3x2 has the largest Quantity in The United States Of America and least amount of Quantity in China
- from the above graph we can also analyze that United States of America has the most number of Quantities for all the CPU units.
- CPU:LGA2011-3x2 and CPU:XEON D-1541 has the maximum amount of quantity in United States of America.

15.DESCRIPTION - Detail 1 LIST_PRICE AS PER COUNTRY

```
In [10]: A=Data.groupby(["DESCRIPTION - Detail 1","COUNTRY_NAME"], as_index=False)[['LIST_PRICE']].sum()  
sns.barplot(data=A,y="DESCRIPTION - Detail 1",x="LIST_PRICE",hue="COUNTRY_NAME")  
plt.title("DESCRIPTION - Detail 1 LIST_PRICE AS PER COUNTRY")  
plt.show()
```

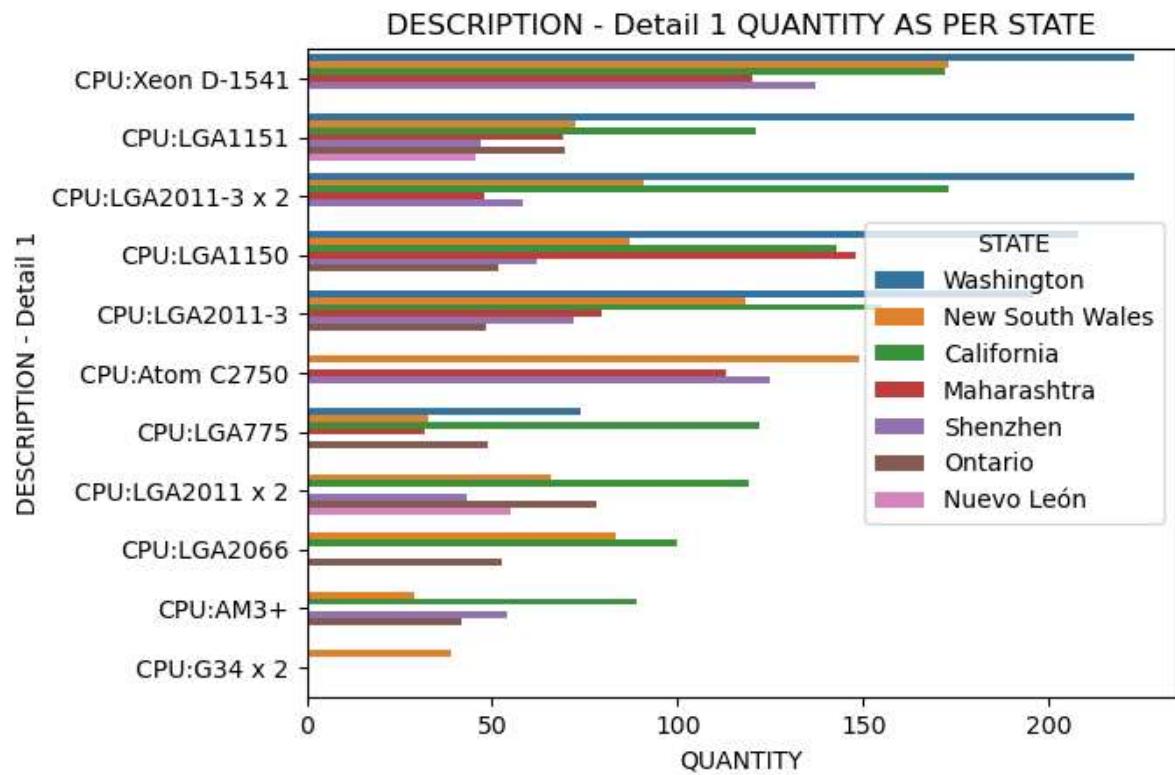


Observation:

- The above graph shows us the List price of products CPU UNITS as per country.
- In the above graph we can see that CPU:XEON D-1541 is equally priced in all the countries like United States of America,Australia and in India.
- We can also analyze that some CPU units are more expensive in India as compare to United States Of America.

16.DESCRIPTION - Detail 1 QUANTITY AS PER STATE

```
In [45]: A=Data.groupby(["DESCRIPTION - Detail 1","STATE"], as_index=False)[['QUANTITY']]
sns.barplot(data=A,y="DESCRIPTION - Detail 1",x="QUANTITY",hue="STATE")
plt.title("DESCRIPTION - Detail 1 QUANTITY AS PER STATE")
plt.show()
```

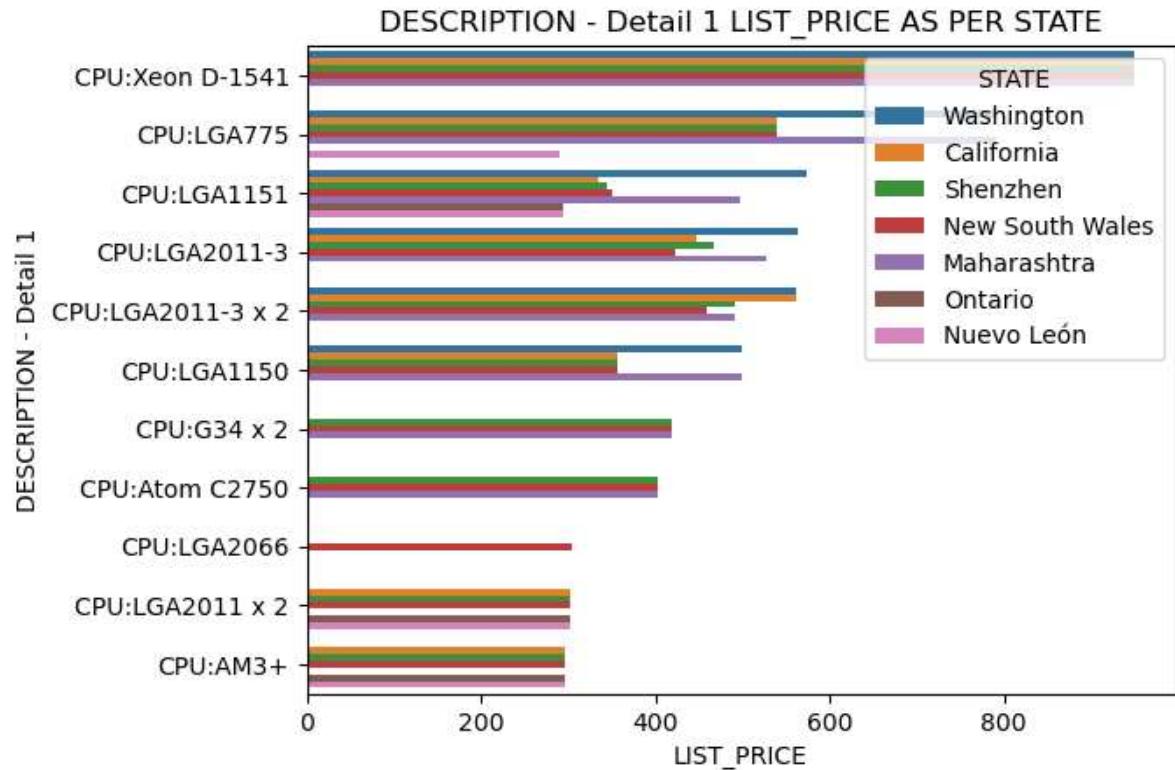


Observation:

- In this above graph it shows us the detailed description of products CPU units quantity as per State.
- We can see in the above graph that CPU:XEON D-1541 has the largest quantity in the state of Washington.
- As we can see that CPU:XEON D-1541 has the largest quantities in top 5 States.
- CPU:XEON D-1541,CPU:LGA1151,CPU:LGA2011-3x2 these three CPU units also have the largest quantities in the state of Washington.

17.DESCRIPTION - Detail 1 LIST_PRICE AS PER STATE

```
In [13]: A=Data.groupby(["DESCRIPTION - Detail 1","STATE"], as_index=False)[['LIST_PRICE']]
sns.barplot(data=A,y="DESCRIPTION - Detail 1",x="LIST_PRICE",hue="STATE")
plt.title("DESCRIPTION - Detail 1 LIST_PRICE AS PER STATE")
plt.show()
```

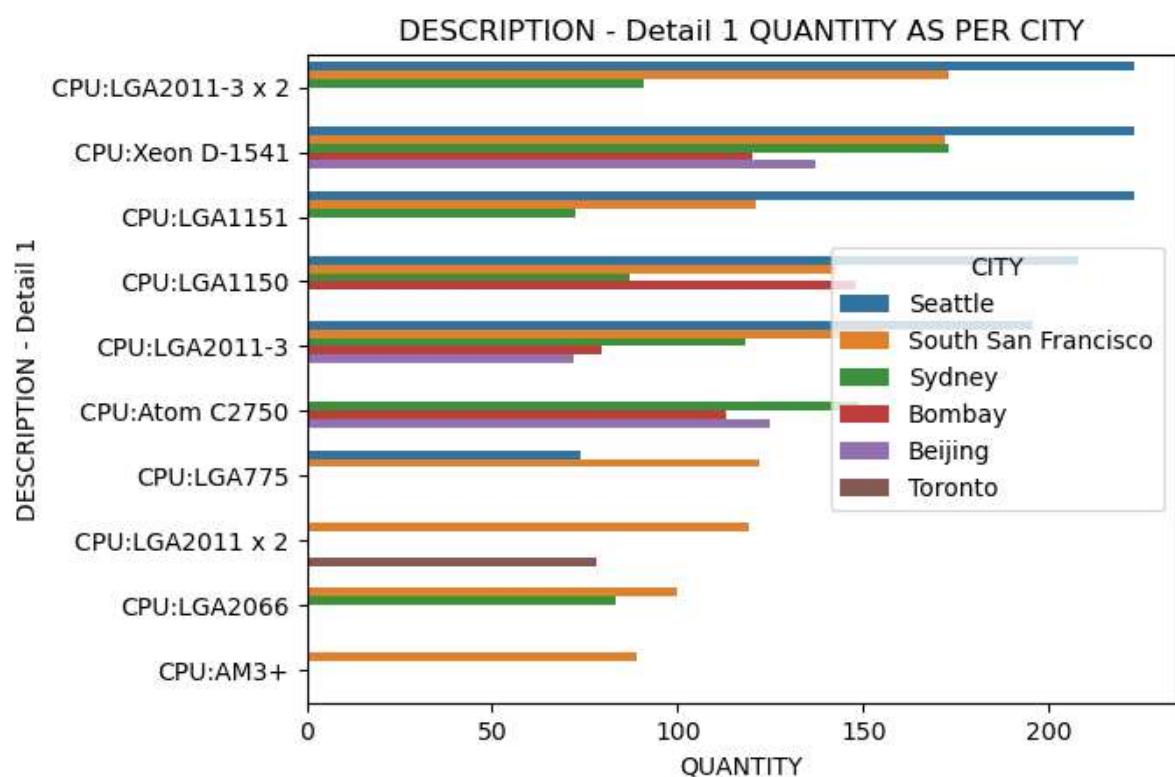


Observation

- The above graph shows us the List price of products CPU UNITS as per State.
- As we can see that CPU:XEON D-1541 has one of the highest price as compare to other units.
- In the above graph we can that CPU:XEON D-1541 is equally priced in top 5 State.

18.DESCRIPTION - Detail 1 Quantity as per City

```
In [14]: A=Data.groupby(["DESCRIPTION - Detail 1","CITY"], as_index=False)[['QUANTITY']].sns.barplot(data=A,y="DESCRIPTION - Detail 1",x="QUANTITY",hue="CITY")plt.title("")plt.show()
```

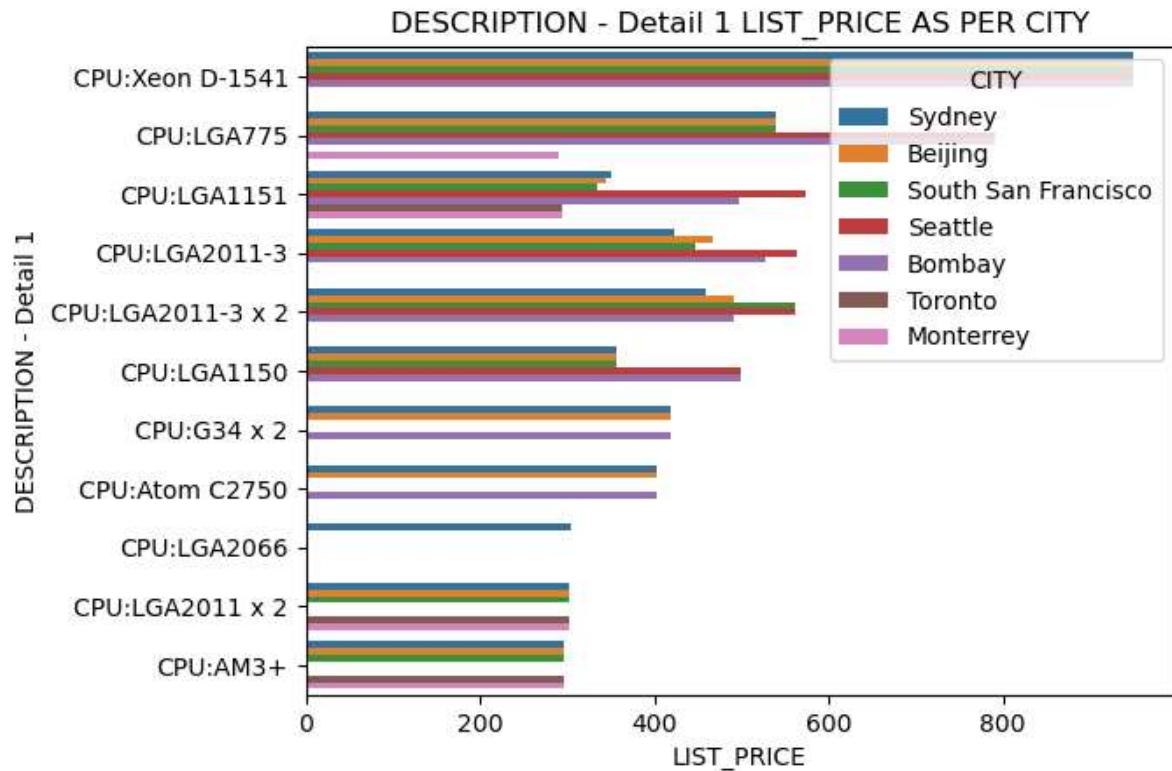


Observation:

- In the above barplot graph it shows the detailed description products Quantity as per city.
- From the above graph we can see that Seattle has the largest Quantity of CPU:LGA2011-3x2.
- From the graph we can see that Seattle has the largest and same quantity of CPU:LGA2011-3x2,CPU:XEON D-1541 and CPU:LGA1151.
- We can also analyze that South San Francisco is the only city that has all the CPU units available.

19.DESCRIPTION - Detail 1 LIST_PRICE AS PER CITY

```
In [46]: A=Data.groupby(["DESCRIPTION - Detail 1","CITY"], as_index=False)[['LIST_PRICE']]
sns.barplot(data=A,y="DESCRIPTION - Detail 1",x="LIST_PRICE",hue="CITY")
plt.title("DESCRIPTION - Detail 1 LIST_PRICE AS PER CITY")
plt.show()
```

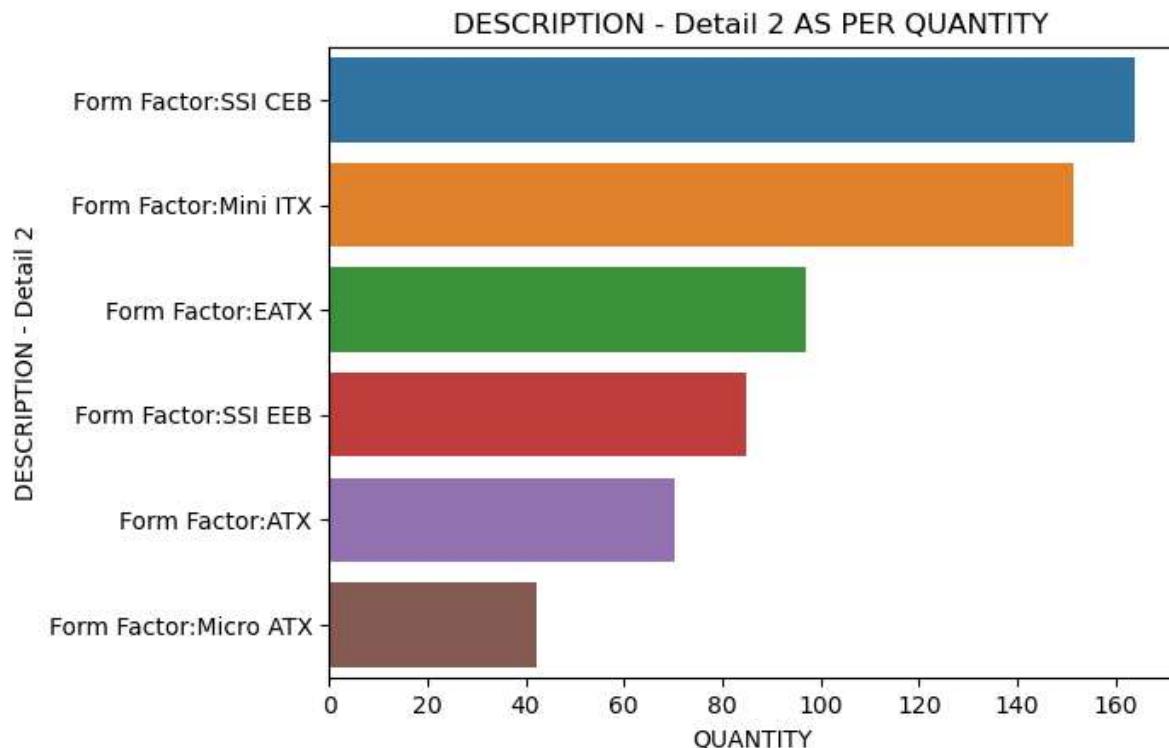


Observation:

- The above graph shows us the List price of products CPU UNITS as per City.
- As we can see that CPU:XEON D-1541 has one of the highest price as compare to other units.
- In the above graph we can see that CPU:XEON D-1541 is equally priced in top5 city.
- As per the graph we can see that CPU:LGA2011-3x2 has equal list price in South San Francisco and seattle and as per last graph CPU:LGA2011-3x2 also has the largest quantity so we can say that this unit is highly in demand in these two cities.

20.DESCRIPTION - Detail 2 AS PER QUANTITY

```
In [71]: A=Data.groupby(["DESCRIPTION - Detail 2"], as_index=False)[ 'QUANTITY'].mean()
sns.barplot(data=A,y="DESCRIPTION - Detail 2",x="QUANTITY")
plt.title("DESCRIPTION - Detail 2 AS PER QUANTITY")
plt.show()
```



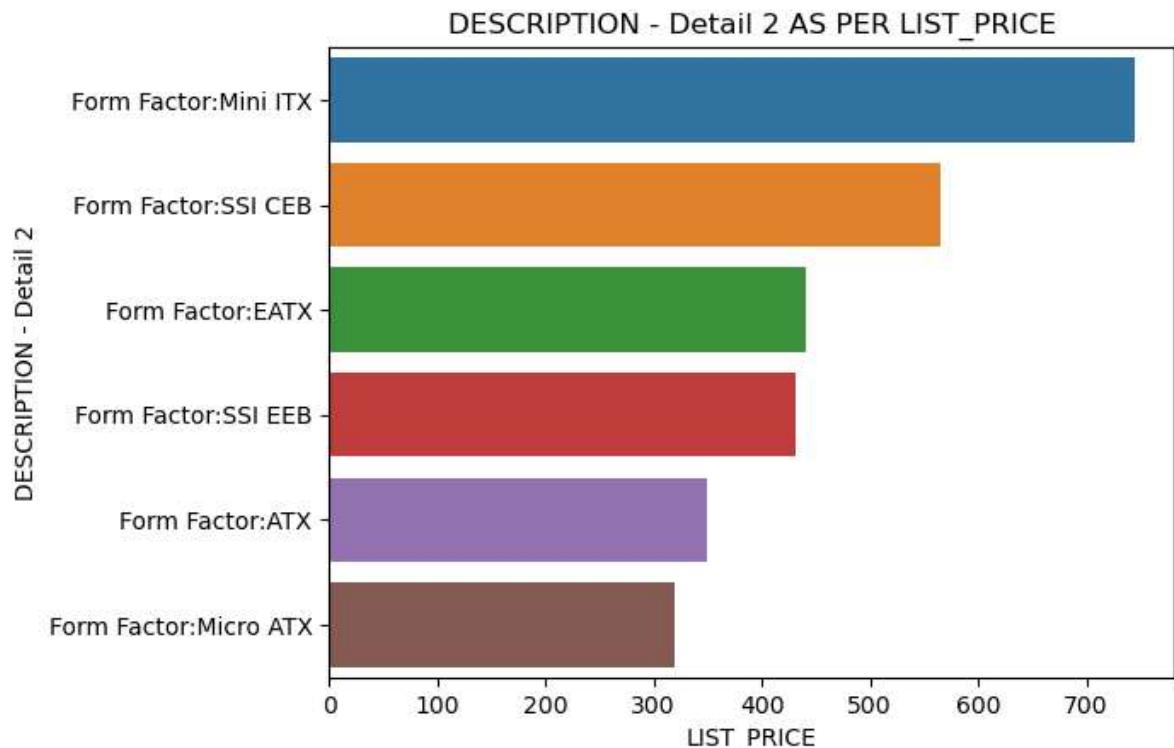
Oservation:

- The above graph shows the detailed description of one of the major components of a mother board as per Quantity.
- From the above Graph we can see that component named as FORM FACTOR:SSI CEB has the maximum Quantity.
- Component named as FORM FACTOR:MINI ITX being the second most component with maximum Quantity.

```
In [ ]:
```

21.DESCRIPTION - Detail 2

```
In [16]: A=Data.groupby(["DESCRIPTION - Detail 2"], as_index=False)[ 'LIST_PRICE'].mean()
sns.barplot(data=A,y="DESCRIPTION - Detail 2",x="LIST_PRICE")
plt.title("DESCRIPTION - Detail 2 AS PER LIST_PRICE")
plt.show()
```

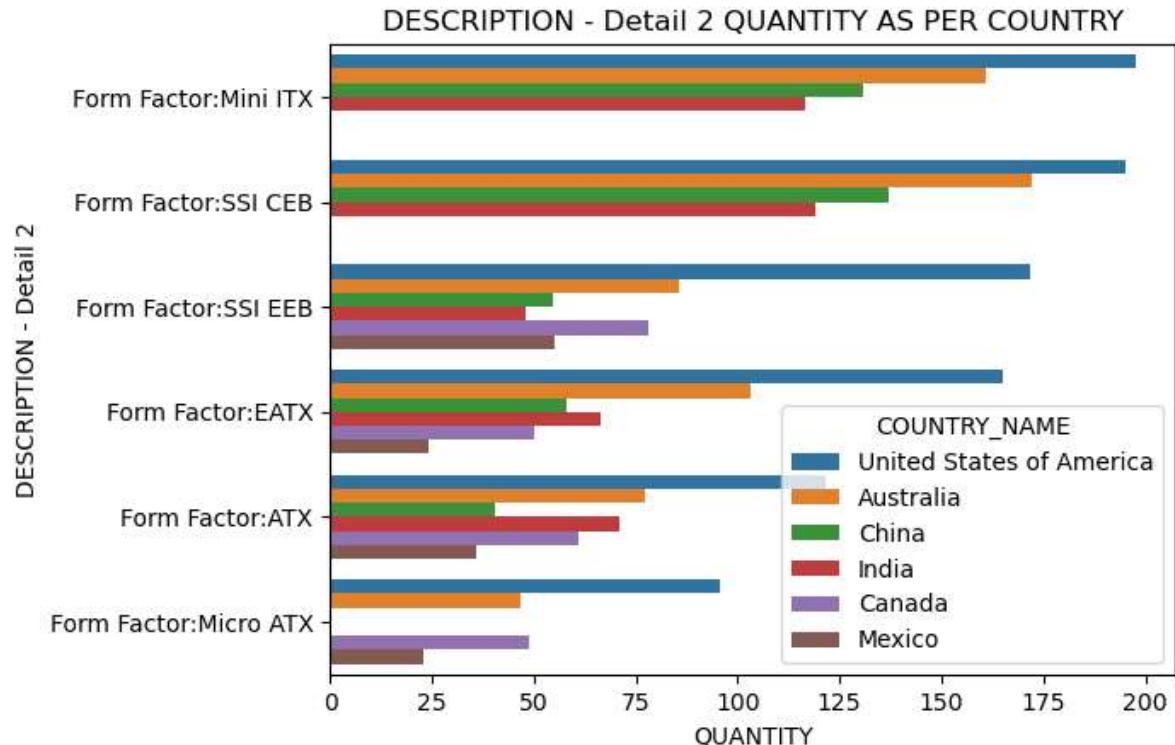


Observation:

- The above graph shows the detailed description of one of the major components of a mother board as per List price.
- From the graph we can see that FORM FACTOR:MINI ITX is the most expensive component among others.
- From the graph component named FORM FACTOR:SSI CEB is the second most expensive component among others.
- As per last graph we saw that component FORM FACTOR:SSI CEB has the largest quantity but in this graph we have seen that its list price is comparatively less than FACTOR:MINI ITX.

22.DESCRIPTION - Detail 2 QUANTITY AS PER COUNTRY

```
In [17]: A=Data.groupby(["DESCRIPTION - Detail 2","COUNTRY_NAME"], as_index=False)[['QUA
sns.barplot(data=A,y="DESCRIPTION - Detail 2",x="QUANTITY",hue="COUNTRY_NAME")
plt.title("DESCRIPTION - Detail 2 QUANTITY AS PER COUNTRY")
plt.show()
```

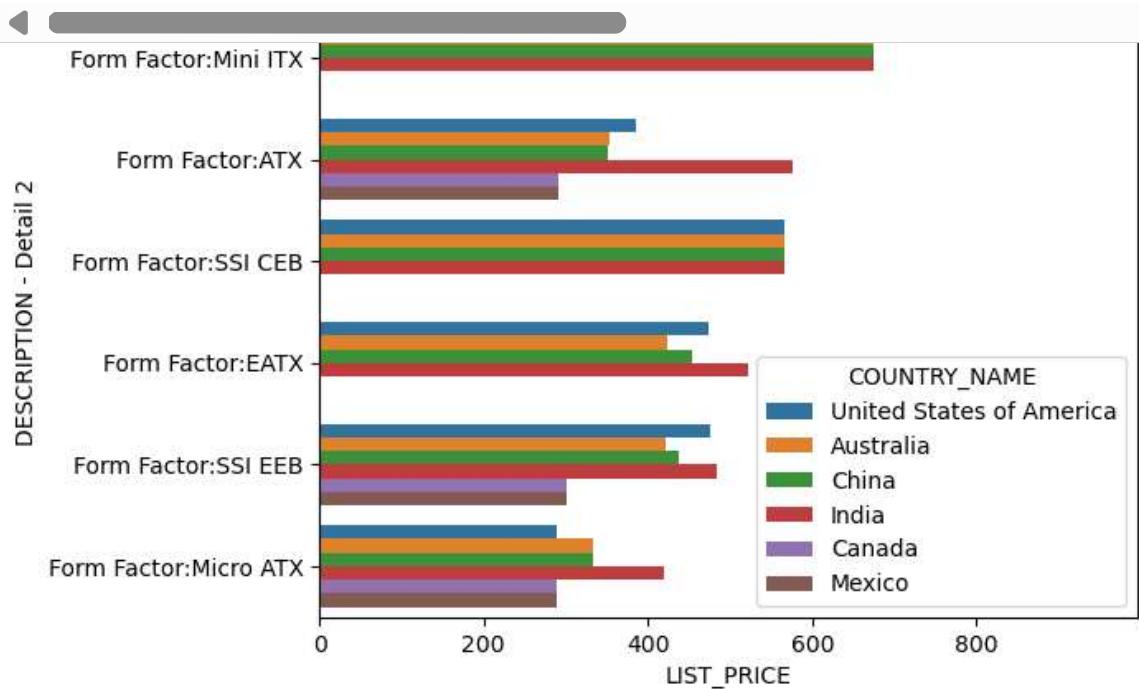


Observation:

- The above graph shows the detailed description of one of the major components of a mother board Quantity as country.
- From the above graph we can see that component FORM FACTOR:MINI ITX and FORM FACTOR:SSI CEB has the maximum quantity in top4 countries.
- United States of America is the only country which has all the major quantity of all the components present with them and is highly in demand.

23.DESCRIPTION - Detail 2 LIST_PRICE AS PER COUNTRY

```
In [21]: A=Data.groupby(["DESCRIPTION - Detail 2","COUNTRY_NAME"], as_index=False)[['LIST_PRICE']].groupby("DESCRIPTION - Detail 2").sum().reset_index()
sns.barplot(data=A,y="DESCRIPTION - Detail 2",x="LIST_PRICE",hue="COUNTRY_NAME")
plt.title("DESCRIPTION - Detail 2 LIST_PRICE AS PER COUNTRY")
plt.show()
```

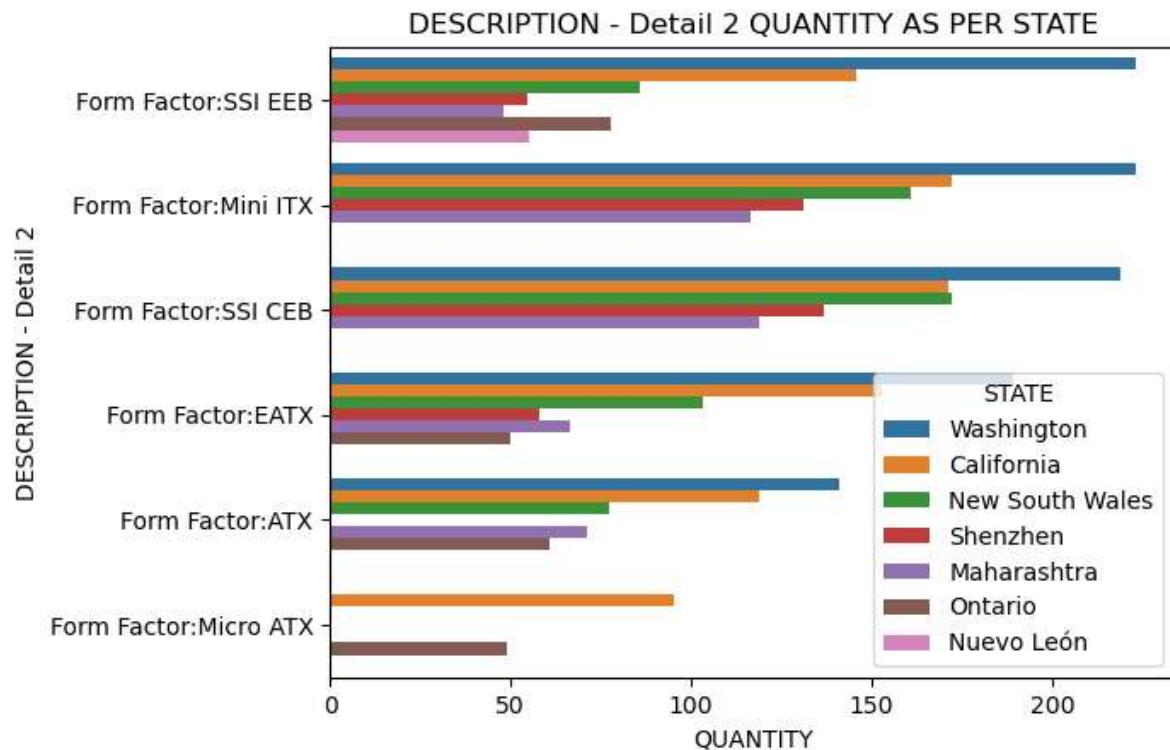


Observation:

- The above graph shows the detailed description of one of the major components of a mother board list price as per country.
- From the graph we can see that FORM FACTOR:MINI ITX is the expensive component in top4 countires where United States of America being the most expensive country.
- From the graph we can also see that FORM FACTOR:SSI CEB is the third most expensive but it is equally priced in the top4 countries.

24.DESCRIPTION - Detail 2 QUANTITY AS PER STATE

```
In [22]: A=Data.groupby(["DESCRIPTION - Detail 2","STATE"], as_index=False)[['QUANTITY']]  
sns.barplot(data=A,y="DESCRIPTION - Detail 2",x="QUANTITY",hue="STATE")  
plt.title("DESCRIPTION - Detail 2 QUANTITY AS PER STATE")  
plt.show()
```

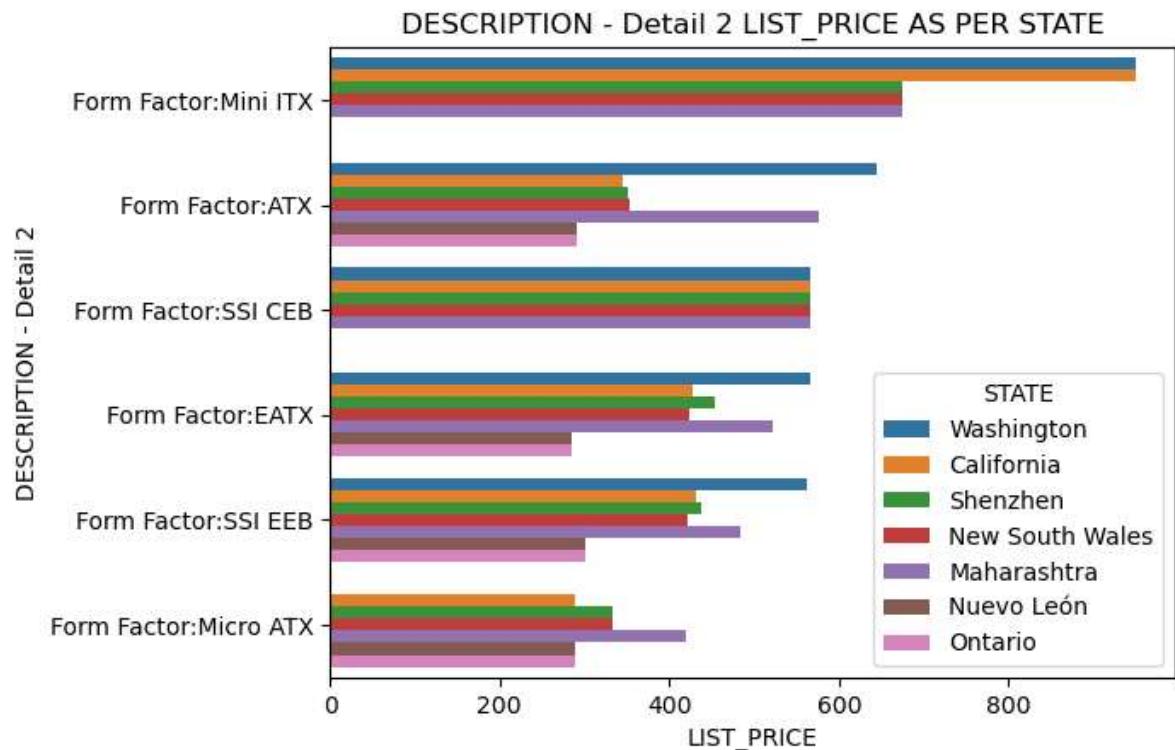


Observation:

- The above graph shows the detailed description of one of the major components of a mother board Quantity as per State.
- From the graph we can see that FORM FACTOR:SSI EEB has the maximum quantity and is available in all states.
- In this graph we can see that FORM FACTOR:MINI ITX and FORM FACTOR:SSI CEB slips down to second and third position.
- From this graph we can also see that almost all the components are in highest quantity in Washington.

25.DESCRIPTION - Detail 2 QUANTITY AS PER STATE

```
In [23]: A=Data.groupby(["DESCRIPTION - Detail 2","STATE"], as_index=False)[['LIST_PRICE']]  
sns.barplot(data=A,y="DESCRIPTION - Detail 2",x="LIST_PRICE",hue="STATE")  
plt.title("DESCRIPTION - Detail 2 LIST_PRICE AS PER STATE")  
plt.show()
```

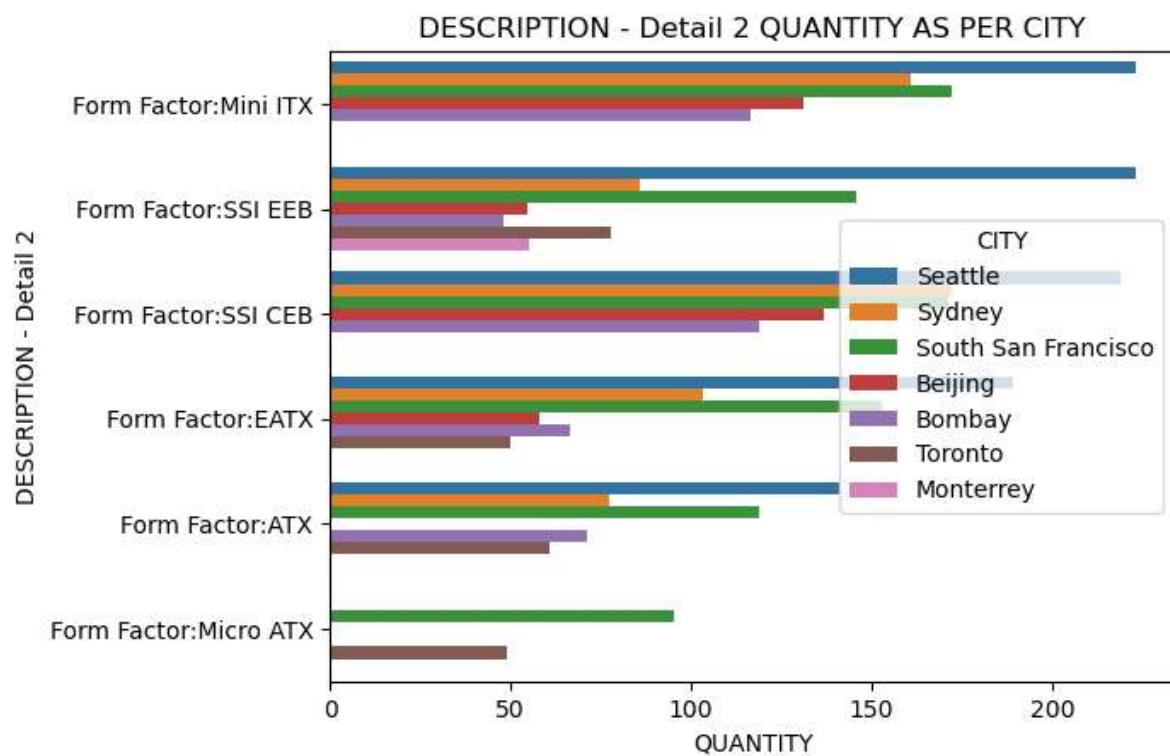


Observation:

- The above graph shows the detailed description of one of the major components of a mother board list price as per state.
- From the graph we can analyze that Form Factor:Mini ITX is expensive where states like Washington and California is highly priced.
- In the above graph we can see that Form Factor:SSI CEB has equal priced component in top5 state.
- As per the Graph Washington,California, Shenzhen,New South Whales and Maharashtra are the only states that has all the marjor components available and are equally priced.
- From the above graph we can see that Washinton is the most expensive state.

26.DESCRIPTION - Detail 2 QUANTITY AS PER CITY

```
In [25]: A=Data.groupby(["DESCRIPTION - Detail 2","CITY"], as_index=False)[['QUANTITY']].sns.barplot(data=A,y="DESCRIPTION - Detail 2",x="QUANTITY",hue="CITY")plt.title("DESCRIPTION - Detail 2 QUANTITY AS PER CITY")plt.show()
```

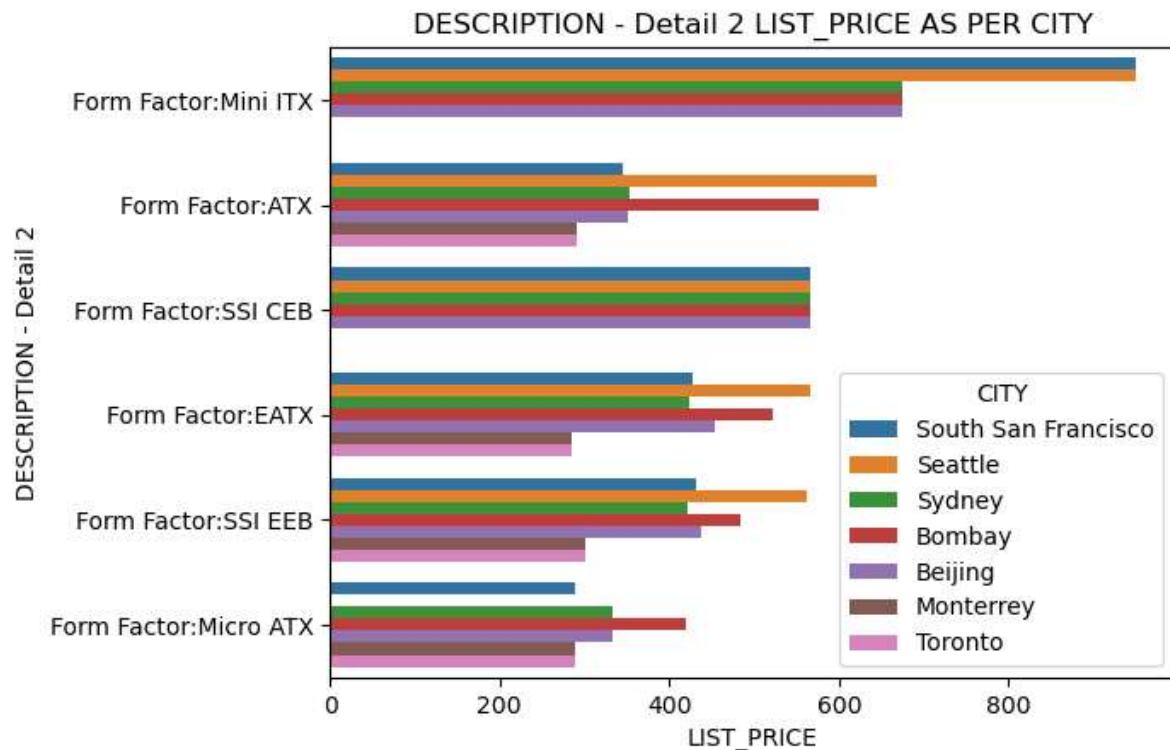


Observation:

- The above graph shows the detailed description of one of the major components of a mother board Quantity as per City.
- From the above graph we can see that Form Factor Mini ITX has the largest Quantity in top5 Cities.
- As per the above graph we can see that the Seattle city has the maximum quantity of top 3 components.

27.DESCRIPTION - Detail 2 LIST_PRICE AS PER CITY

```
In [27]: A=Data.groupby(["DESCRIPTION - Detail 2","CITY"], as_index=False)[['LIST_PRICE']]  
sns.barplot(data=A,y="DESCRIPTION - Detail 2",x="LIST_PRICE",hue="CITY")  
plt.title("DESCRIPTION - Detail 2 LIST_PRICE AS PER CITY")  
plt.show()
```

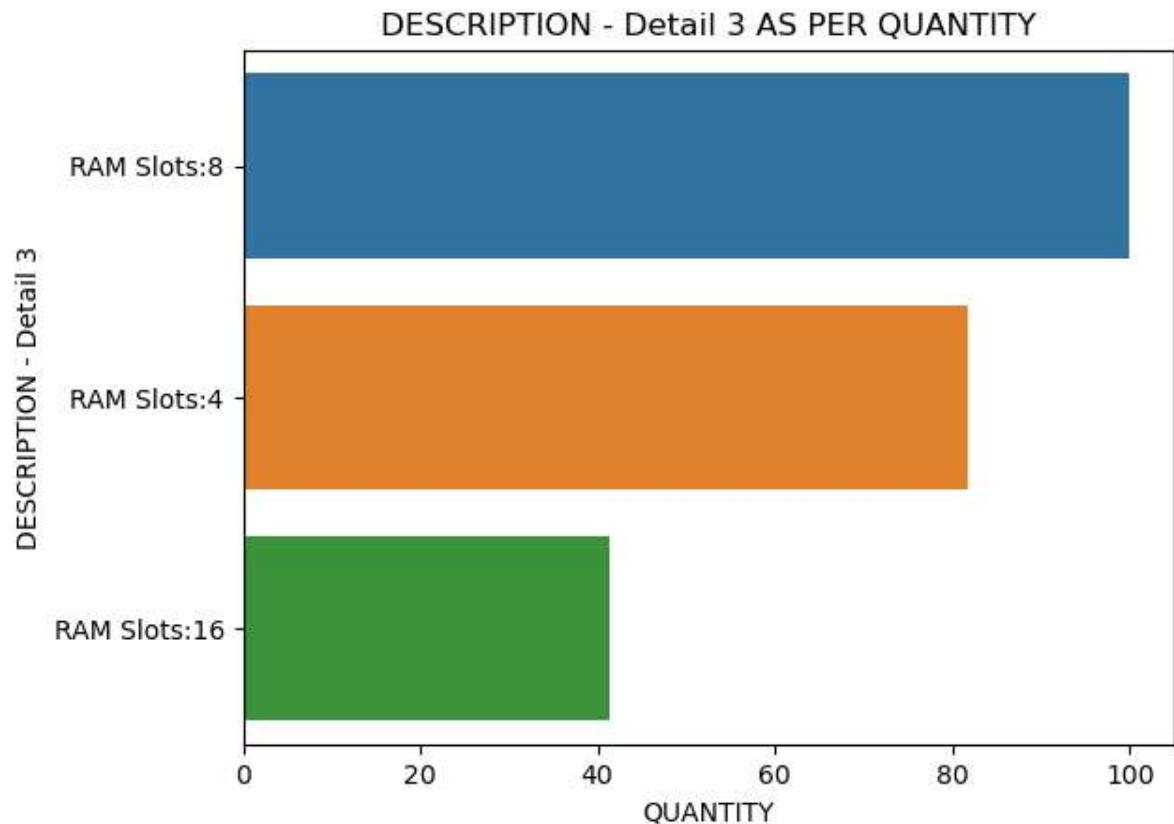


Observation:

- The above graph shows the detailed description of one of the major components of a mother board list price as per City.
- From the graph we can analyze that Form Factor:Mini ITX is expensive but is highly expensive in cities like South San Francisco.
- In the above graph we can see that Form Factor:SSI CEB is equally priced in top5 cities.
- In the above graph we can see that all the components are mostly present in all the cities.

28.DESCRIPTION - Detail 3

```
In [24]: A=Data.groupby(["DESCRIPTION - Detail 3"], as_index=False)[ 'QUANTITY'].mean().  
sns.barplot(data=A,y="DESCRIPTION - Detail 3",x="QUANTITY")  
plt.title("DESCRIPTION - Detail 3 AS PER QUANTITY")  
plt.show()
```

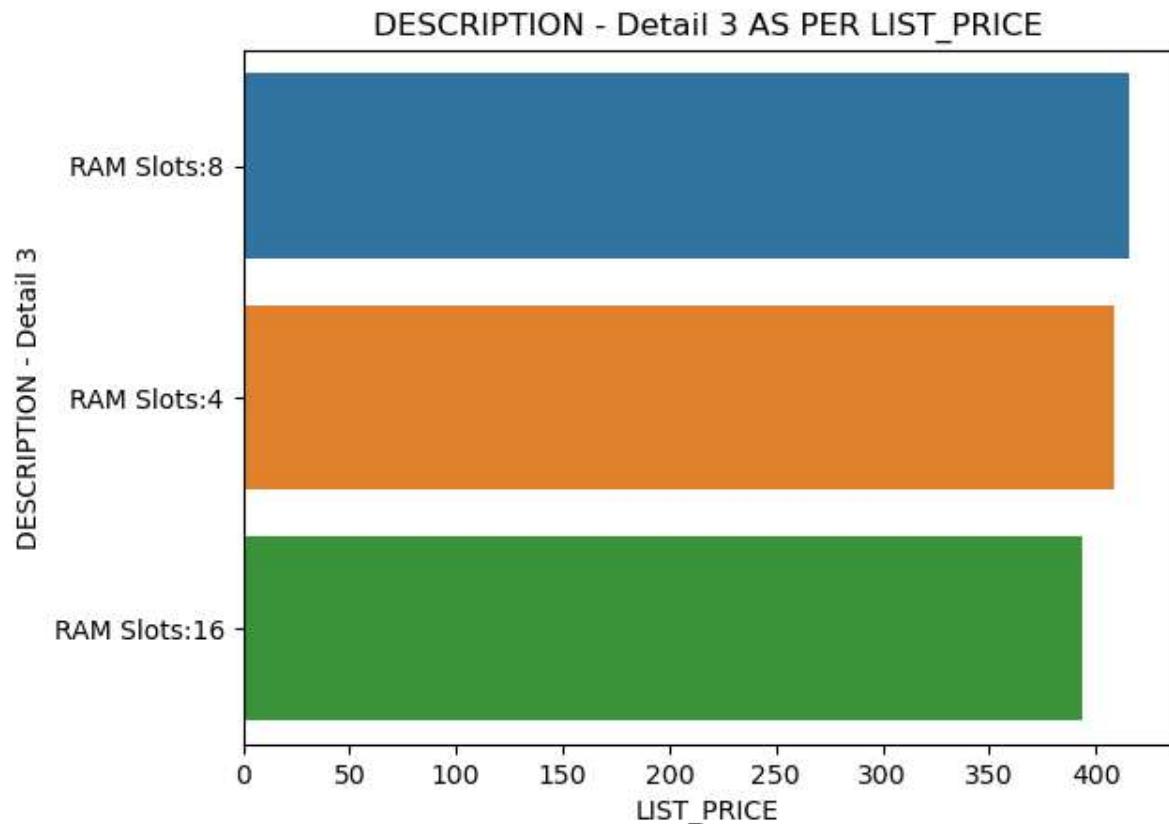


Observation:

- The above graph shows us the Quantity of different RAM slots.
- From the above graph we say that RAM slots:8 has the maximum Quantity.
- RAM Slot: 16 being the minimum quantity.

29.DESCRIPTION - Detail 3 AS PER LIST_PRICE

```
In [28]: A=Data.groupby(["DESCRIPTION - Detail 3"], as_index=False)[ 'LIST_PRICE'].mean()
sns.barplot(data=A,y="DESCRIPTION - Detail 3",x="LIST_PRICE")
plt.title("DESCRIPTION - Detail 3 AS PER LIST_PRICE")
plt.show()
```



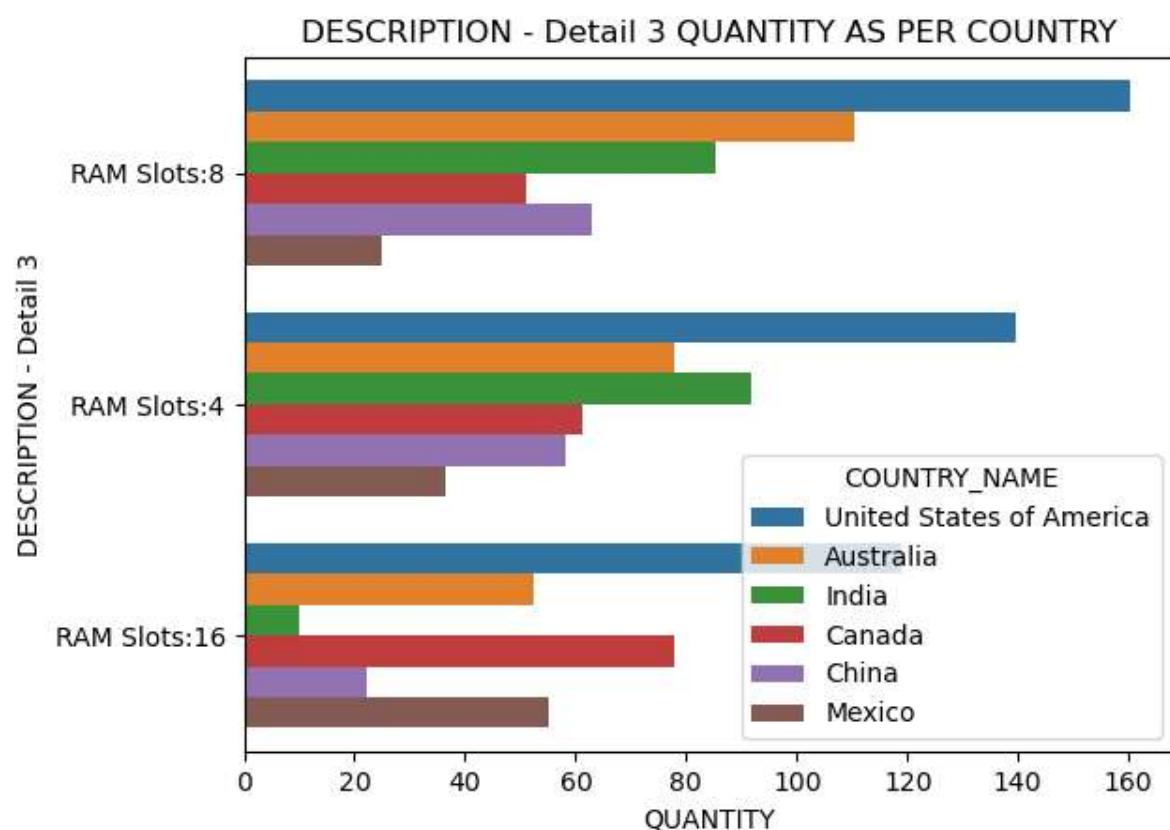
Observation:

- The above graph shows us the list price of different RAM slots.
- From the graph we can see that RAM Slots:8 is expensive.
- From the Above graph we can see that all the RAM Slots are almost equally priced where their cost are mostly around 350-400

30.DESCRIPTION - Detail 3 QUANTITY AS PER COUNTRY

In [29]:

```
A=Data.groupby(["DESCRIPTION - Detail 3","COUNTRY_NAME"], as_index=False)[['QUA  
sns.barplot(data=A,y="DESCRIPTION - Detail 3",x="QUANTITY",hue="COUNTRY_NAME")  
plt.title("DESCRIPTION - Detail 3 QUANTITY AS PER COUNTRY")  
plt.show()
```

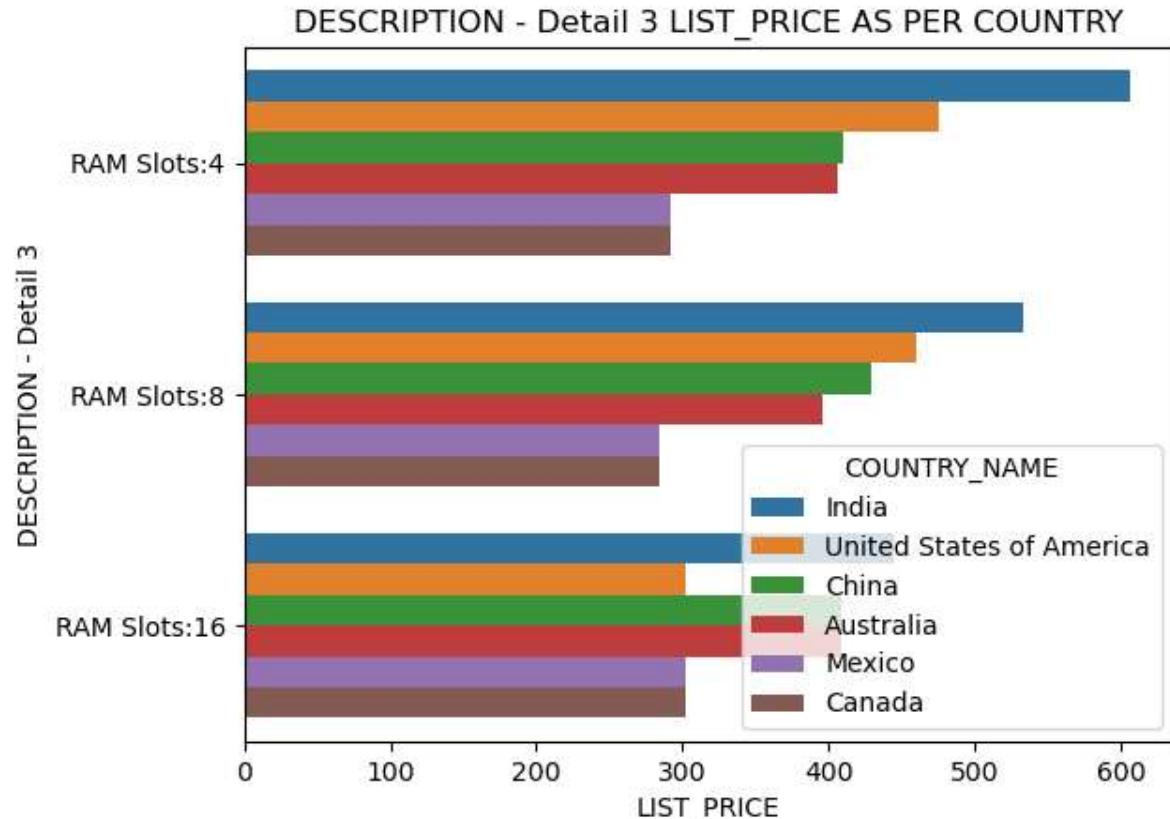


Observation:

- The above graph shows us the Quantity of different RAM slots as per Country.
- From the graph we can say that RAM Slots:8 has the most quantity compared to other RAM Slots where its maximum Quantity is 160 in United States Of America.
- we can also see that all the RAM Slot: are available in all the countries.
- From the graph we can also see that United States Of America has the maximum quantity of all the RAM Slots.

31.DESCRIPTION - Detail 3 LIST_PRICE AS PER COUNTRY

```
In [30]: A=Data.groupby(["DESCRIPTION - Detail 3","COUNTRY_NAME"], as_index=False)[['LIST_PRICE']].sum()  
sns.barplot(data=A,y="DESCRIPTION - Detail 3",x="LIST_PRICE",hue="COUNTRY_NAME")  
plt.title("DESCRIPTION - Detail 3 LIST_PRICE AS PER COUNTRY")  
plt.show()
```

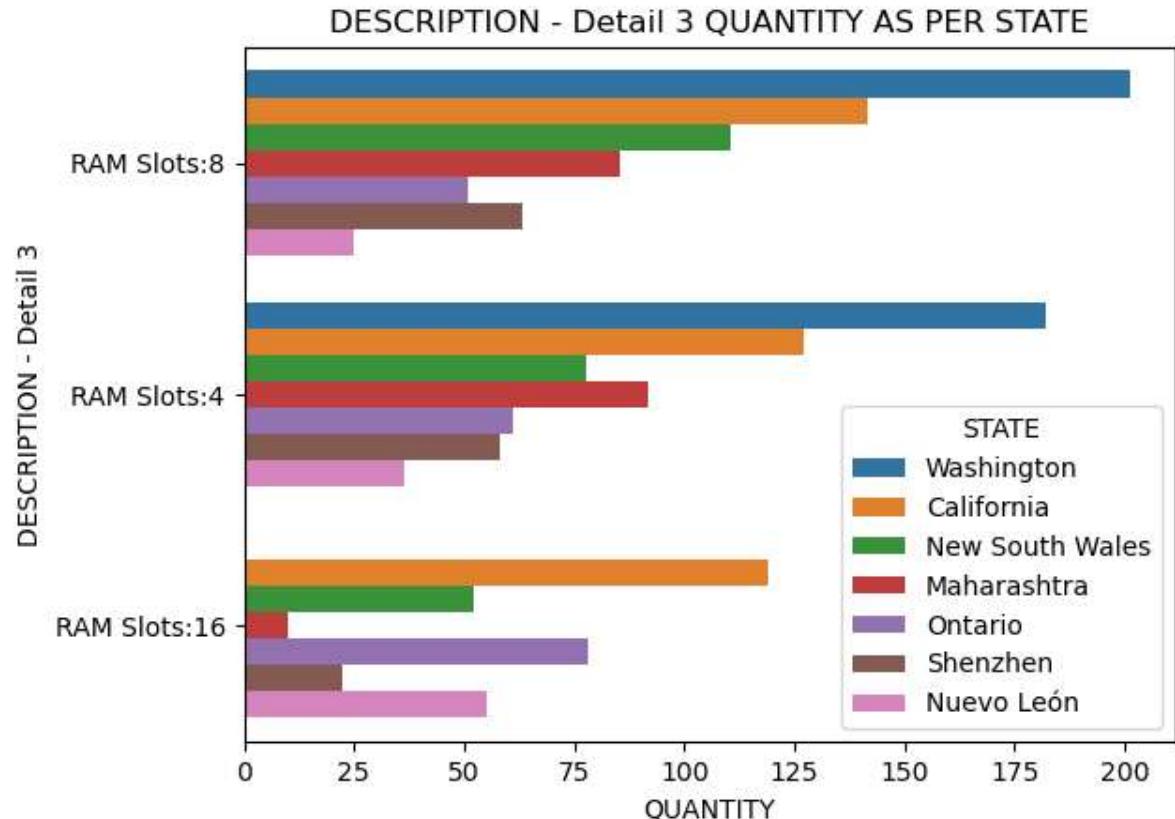


Observation:

- The above graph shows us the List_Price of different RAM slots as per Country.
- In the above graph we can see that RAM Slot:4 is the most priced slots compared to others.
- From the above graph we can see that RAM Slot:4 is most expensive in India where it costs around 600.
- And from the above graph we can see that India is the most expensive country.

32.DESCRIPTION - Detail 3 QUANTITY AS PER STATE"

```
In [31]: A=Data.groupby(["DESCRIPTION - Detail 3","STATE"], as_index=False)[['QUANTITY']]  
sns.barplot(data=A,y="DESCRIPTION - Detail 3",x="QUANTITY",hue="STATE")  
plt.title("DESCRIPTION - Detail 3 QUANTITY AS PER STATE")  
plt.show()
```

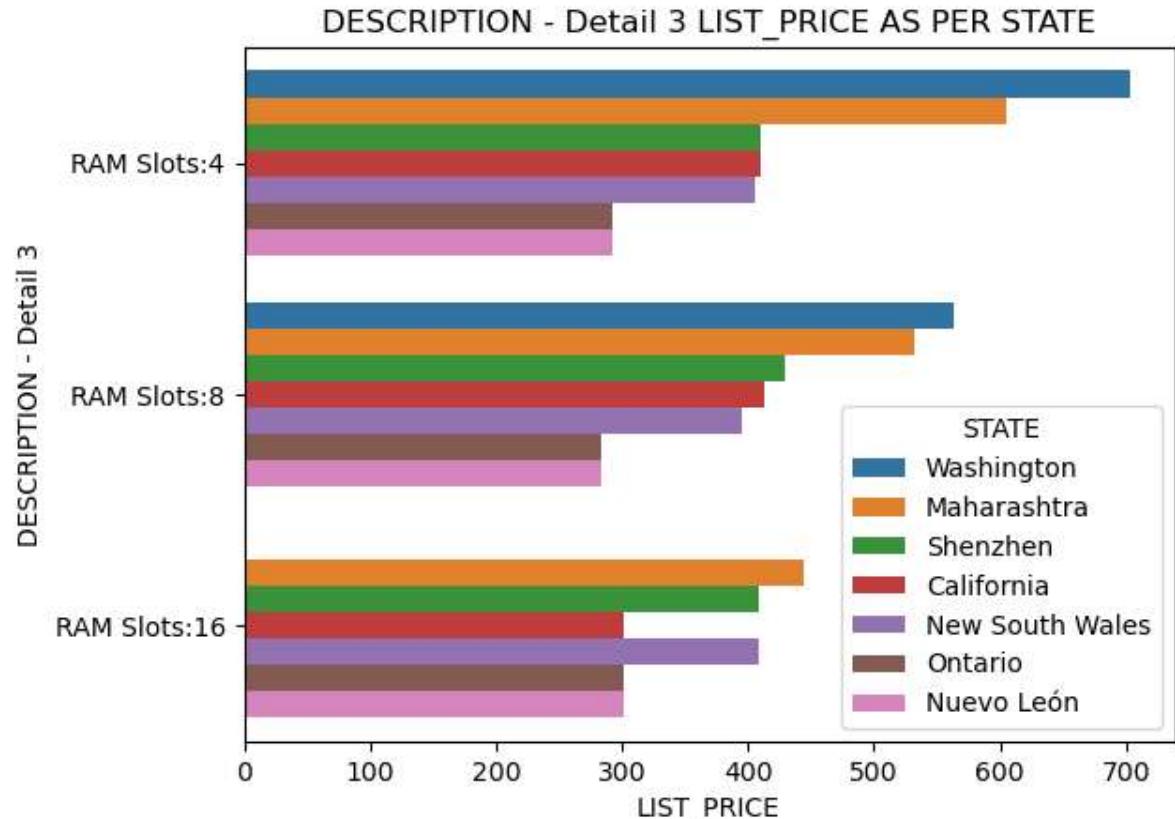


Observation:

- The above graph shows us the Quantity of different RAM slots as per State.
- From the graph we see that RAM Slot:8 has the most quantity.
- In the above graph we can see that Washington has the maximum quantity of RAM Slot:8.
- Washington not only has the highest quantity of RAM Slot:8 but has the highest quantity of RAM Slot:4 also.
- RAM Slot:16 usually has the quantity majorly in states like California.

33.DESCRIPTION - Detail 3 LIST_PRICE AS PER STATE

```
In [32]: A=Data.groupby(["DESCRIPTION - Detail 3","STATE"], as_index=False)[['LIST_PRICE']]
sns.barplot(data=A,y="DESCRIPTION - Detail 3",x="LIST_PRICE",hue="STATE")
plt.title("DESCRIPTION - Detail 3 LIST_PRICE AS PER STATE")
plt.show()
```

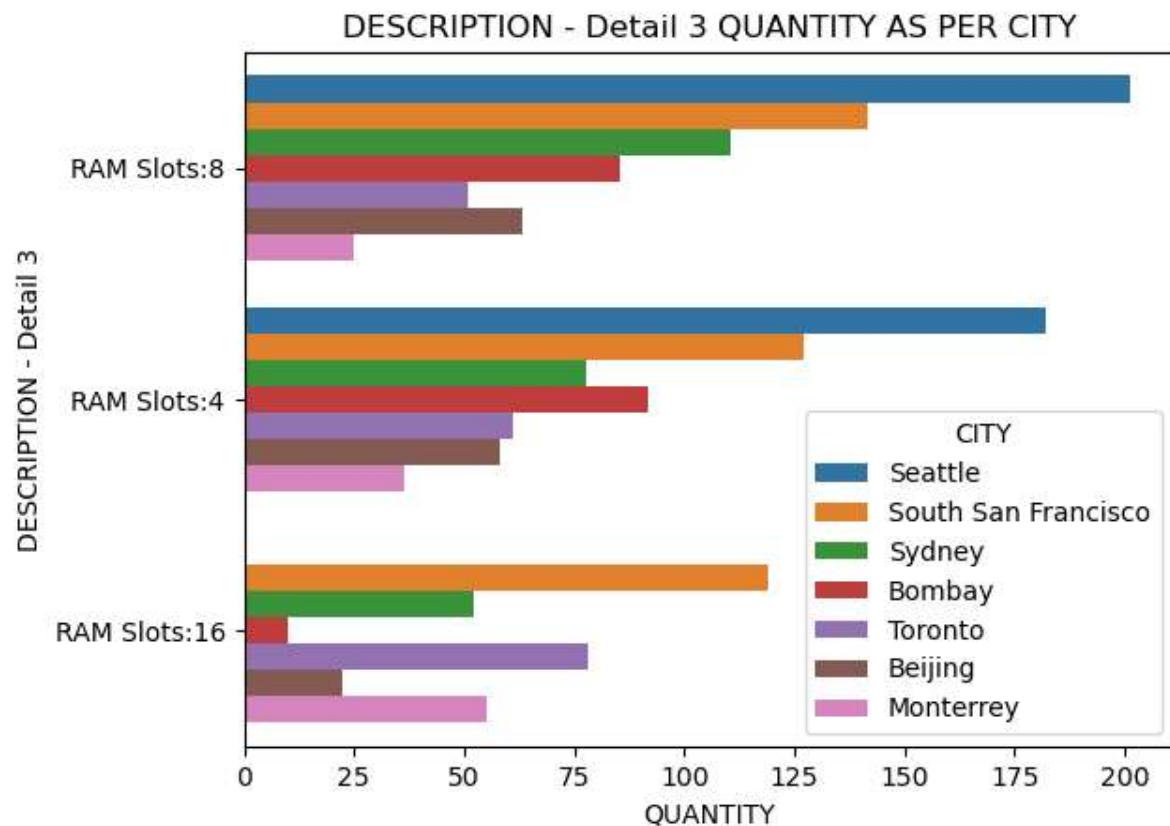


Observation:

- The above graph shows us the List_Price of different RAM slots as per States.
- From the graph we can see that RAM Slot:4 is the most priced Slot.
- As we can see from the graph that Washington is the most expensive state for RAM Slot:4 and RAM Slot8.

34.DESCRIPTION - Detail 3 QUANTITY AS PER CITY

```
In [33]: A=Data.groupby(["DESCRIPTION - Detail 3","CITY"], as_index=False)[['QUANTITY']].sns.barplot(data=A,y="DESCRIPTION - Detail 3",x="QUANTITY",hue="CITY")plt.title("DESCRIPTION - Detail 3 QUANTITY AS PER CITY")plt.show()
```

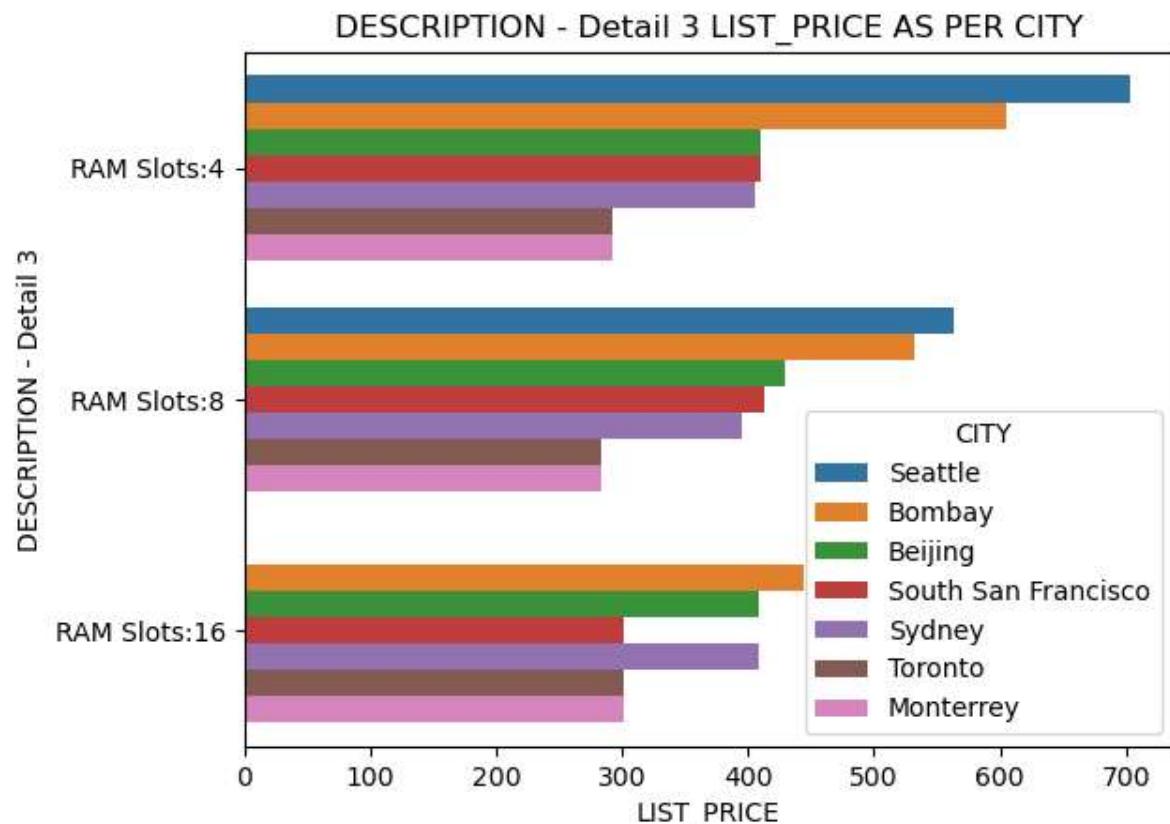


Observation:

- The above graph shows us the Quantity of different RAM slots as per City.
- As per the Graph Seattle hold the most Quantity of RAM Slots:8 and RAM Slot:4.
- Both RAM Slot:4 and RAM Slot:8 is available in all the cities.

35.DESCRIPTION - Detail 3 LIST_PRICE AS PER CITY

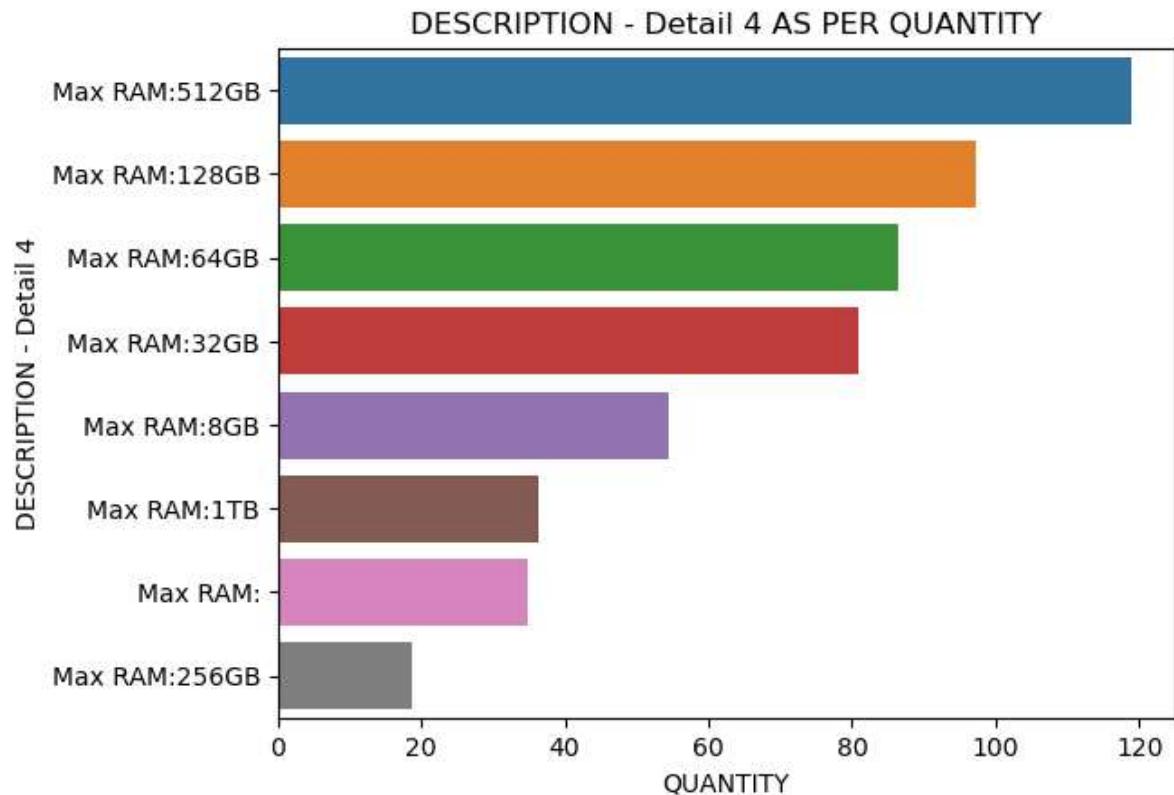
```
In [34]: A=Data.groupby(["DESCRIPTION - Detail 3","CITY"], as_index=False)[['LIST_PRICE']]  
sns.barplot(data=A,y="DESCRIPTION - Detail 3",x="LIST_PRICE",hue="CITY")  
plt.title("DESCRIPTION - Detail 3 LIST_PRICE AS PER CITY")  
plt.show()
```



- The above graph shows us the List_Price of different RAM slots as per City.
- From the graph we can see that RAM Slot:4 is the most priced Slot.
- From the graph we can say that Seattle is the most expensive city for both RAM Slots:4&8.
- Bombay being the most expensive for RAM Slot:16.

36.DESCRIPTION - Detail 4

```
In [73]: A=Data.groupby(["DESCRIPTION - Detail 4"], as_index=False)[ 'QUANTITY'].mean().  
sns.barplot(data=A,y="DESCRIPTION - Detail 4",x="QUANTITY")  
plt.title("DESCRIPTION - Detail 4 AS PER QUANTITY")  
plt.show()
```

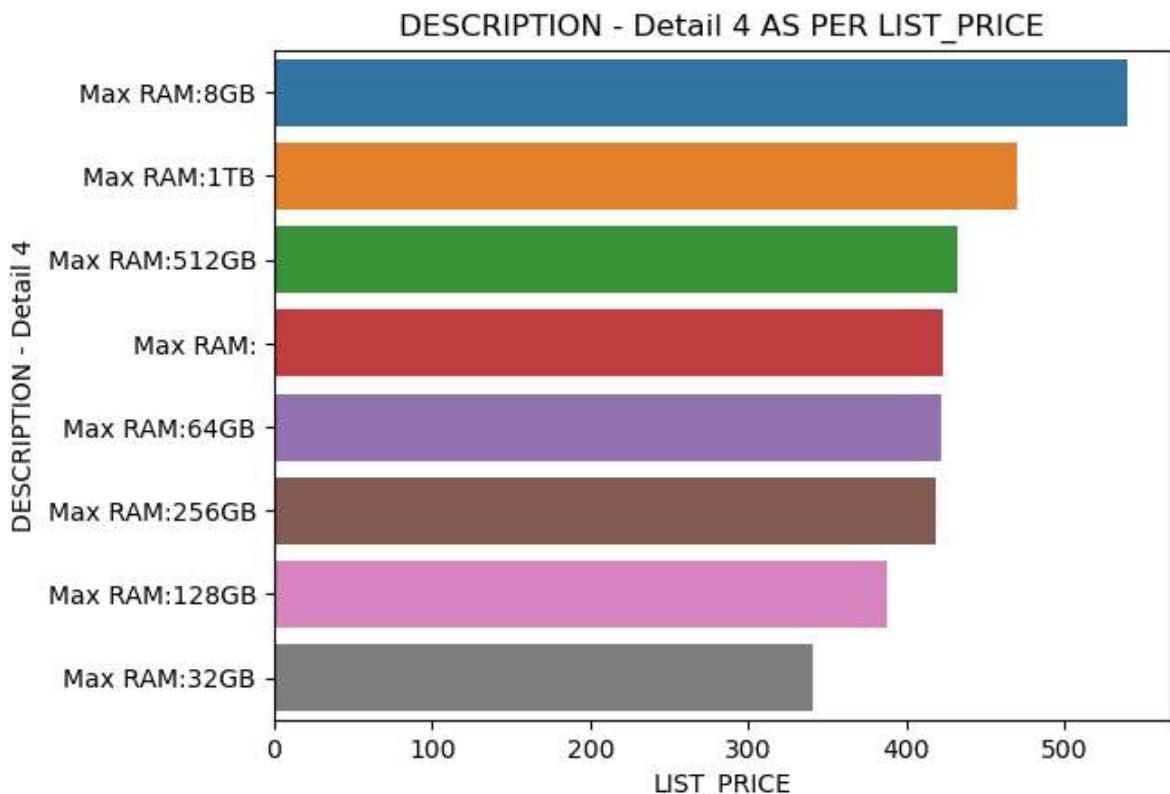


Observation:

- The above graph shows us the Quantity of different MAX RAM .
- From the above graph we can see that MAX RAM with 512GB has the highest Quantity exceeding 100 units
- In the above graph MAX RAM-256GB has tye least quantity only 19 units.

37.DESCRIPTION - Detail 4 AS PER LIST_PRICE

```
In [35]: A=Data.groupby(["DESCRIPTION - Detail 4"], as_index=False)[ 'LIST_PRICE'].mean()
sns.barplot(data=A,y="DESCRIPTION - Detail 4",x="LIST_PRICE")
plt.title("DESCRIPTION - Detail 4 AS PER LIST_PRICE")
plt.show()
```

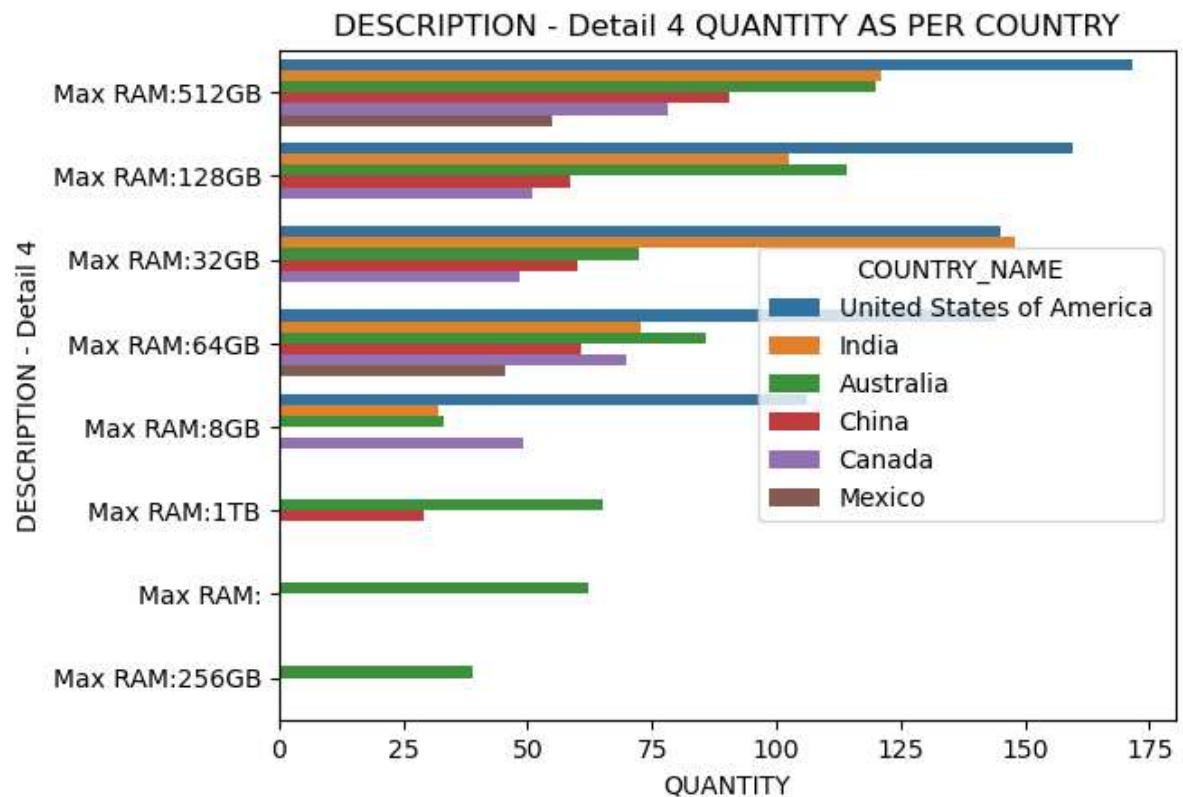


Observation:

- The above graph shows us the List_price of different MAX RAM.
- From the above graph we can see that MAX RAM:8GB has the highest price at 500
- MAX RAM:32GB being the cheapest component at 320

38.DESCRIPTION - Detail 4 QUANTITY AS PER COUNTRY

```
In [41]: A=Data.groupby(["DESCRIPTION - Detail 4","COUNTRY_NAME"], as_index=False)[['QUA
sns.barplot(data=A,y="DESCRIPTION - Detail 4",x="QUANTITY",hue="COUNTRY_NAME")
plt.title("DESCRIPTION - Detail 4 QUANTITY AS PER COUNTRY")
plt.show()
```

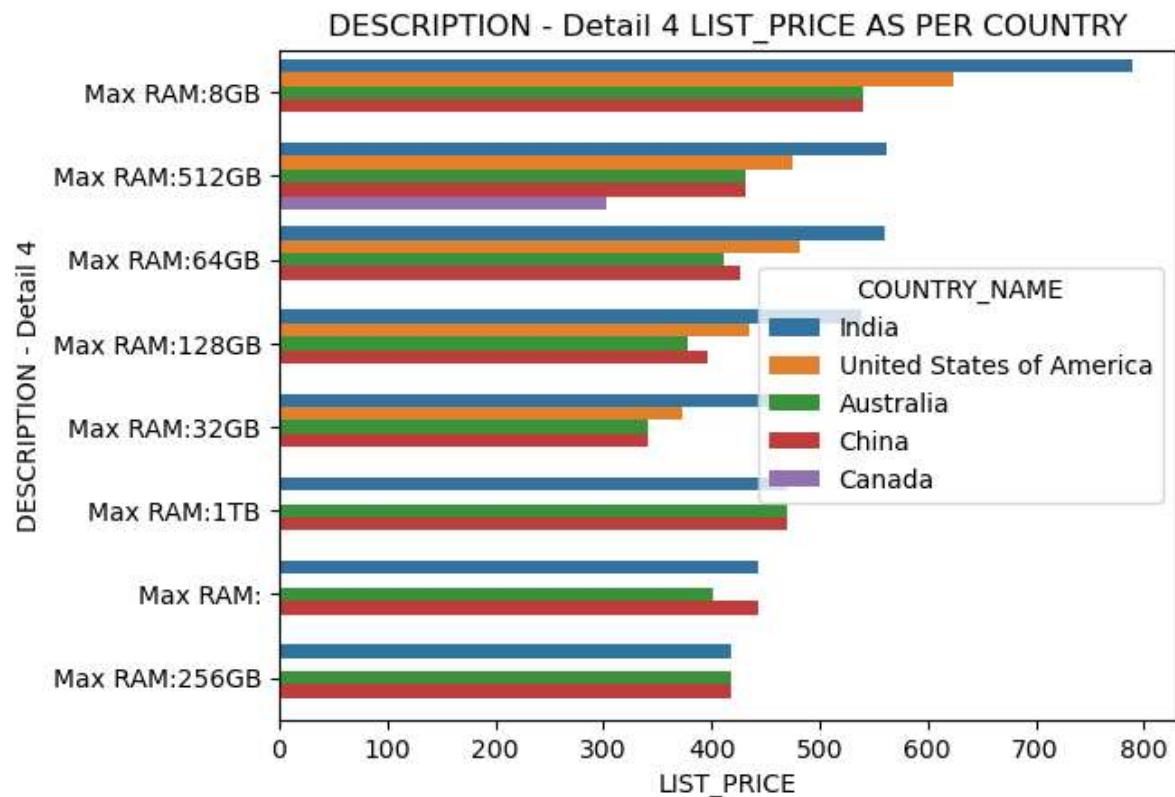


Observation:

- The above graph shows us the Quantity of different MAX RAM as per Country.
- From the above graph we can see that MAX RAM:512GB holds the largest quantity.
- From the graph we can analyze that United States of America holds the maximum quantity of the MAX RAM:512GB.
- In the above graph we can also say that India is being the second country to have maximum quantity of MAX RAM.

39.DESCRIPTION - Detail 4 LIST_PRICE AS PER COUNTRY

```
In [37]: A=Data.groupby(["DESCRIPTION - Detail 4","COUNTRY_NAME"], as_index=False)[['LIST_PRICE']].sum()  
sns.barplot(data=A,y="DESCRIPTION - Detail 4",x="LIST_PRICE",hue="COUNTRY_NAME")  
plt.title("DESCRIPTION - Detail 4 LIST_PRICE AS PER COUNTRY")  
plt.show()
```

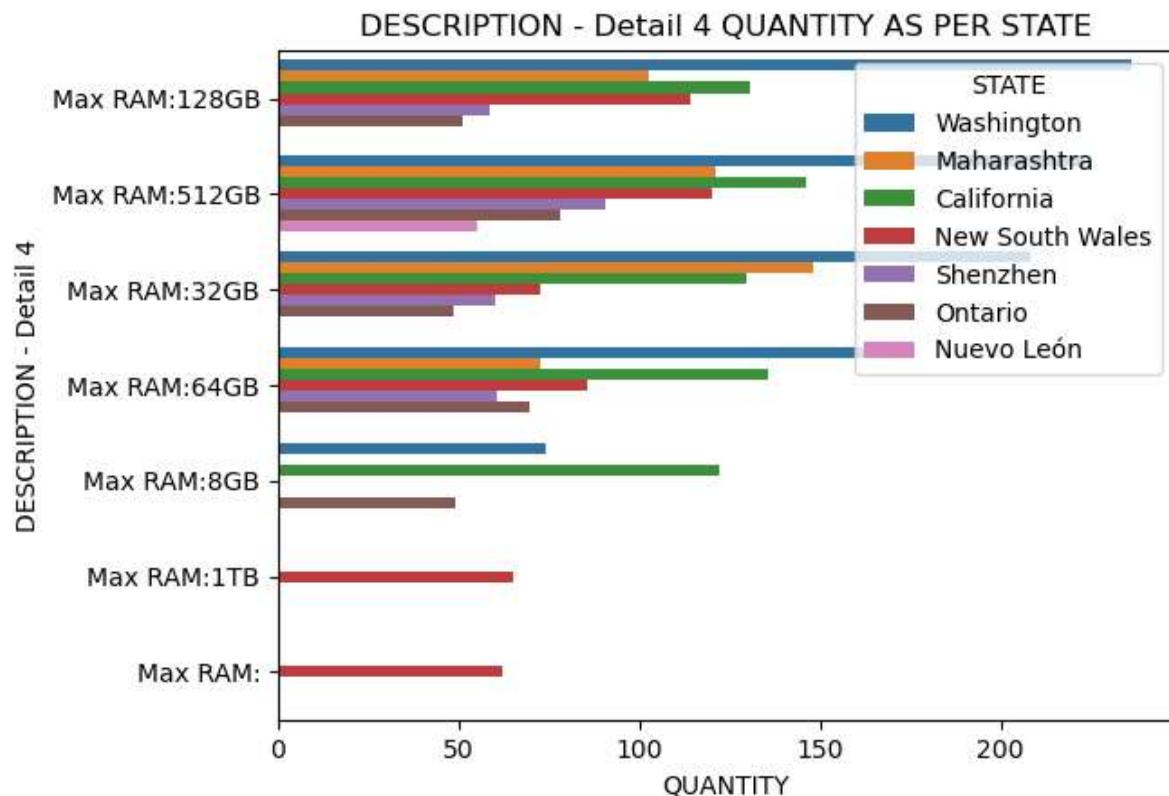


Observation:

- The above graph shows us the List_price of different MAX RAM as per Country.
- From the graph we can see that MAX RAM:8GB is the expensive component compared to other MAX RAM.
- From the above graph we can also see that MAX RAM:8GB is very expensive in India.
- All the MAX RAM are expensive in India and United States of America being second.

40.DESCRIPTION - Detail 4 QUANTITY AS PER STATE

```
In [38]: A=Data.groupby(["DESCRIPTION - Detail 4","STATE"], as_index=False)[['QUANTITY']]  
sns.barplot(data=A,y="DESCRIPTION - Detail 4",x="QUANTITY",hue="STATE")  
plt.title("DESCRIPTION - Detail 4 QUANTITY AS PER STATE")  
plt.show()
```

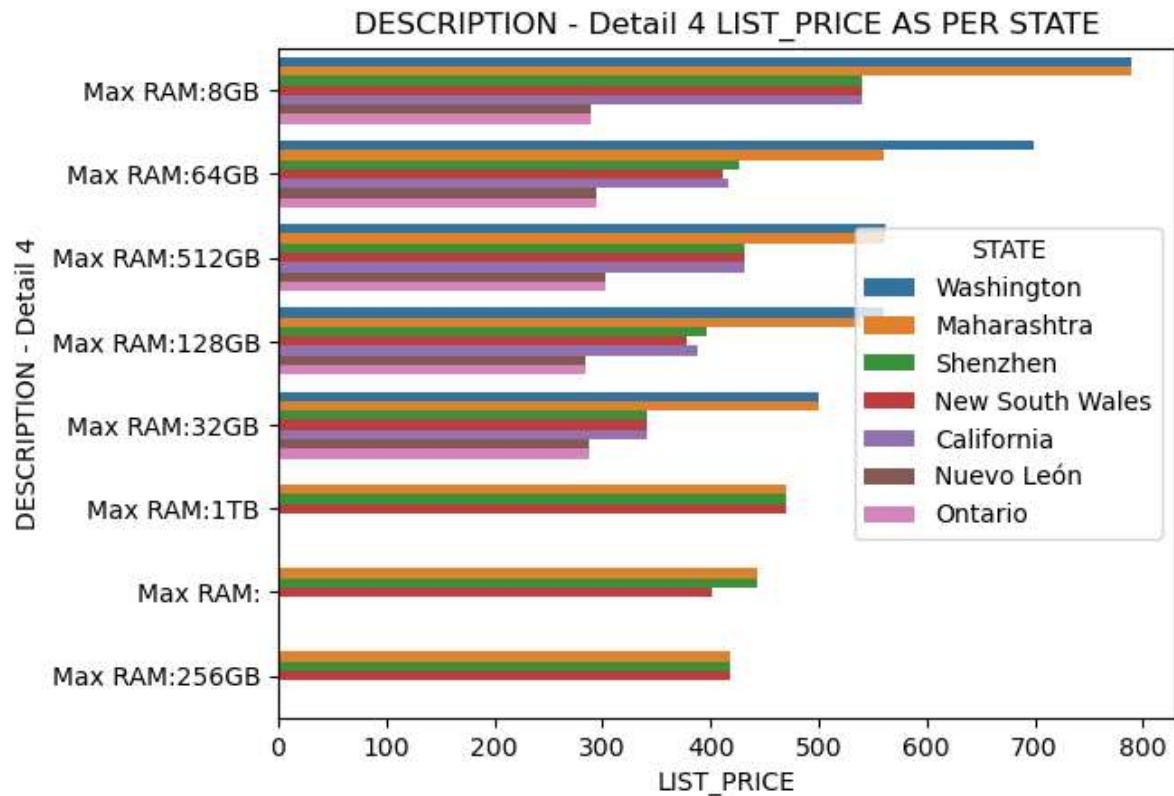


Observation:

- The above graph shows us the Quantity of different MAX RAM as per State.
- From the above graph we can see that MAX RAM:128GB has the highest quantity.
- In the above graph we can see that Washington holds the highest number of quantity in all the MAX RAM processor.
- Only MAX RAM:512GB has the availability in all the states.
- As we can also see that MAX RAM and MAX RAM:1TB is only has its availability in New South Wales

41.DESCRIPTION - Detail 4 LIST_PRICE AS PER STAT

```
In [39]: A=Data.groupby(["DESCRIPTION - Detail 4","STATE"], as_index=False)[['LIST_PRICE']]
sns.barplot(data=A,y="DESCRIPTION - Detail 4",x="LIST_PRICE",hue="STATE")
plt.title("DESCRIPTION - Detail 4 LIST_PRICE AS PER STATE")
plt.show()
```

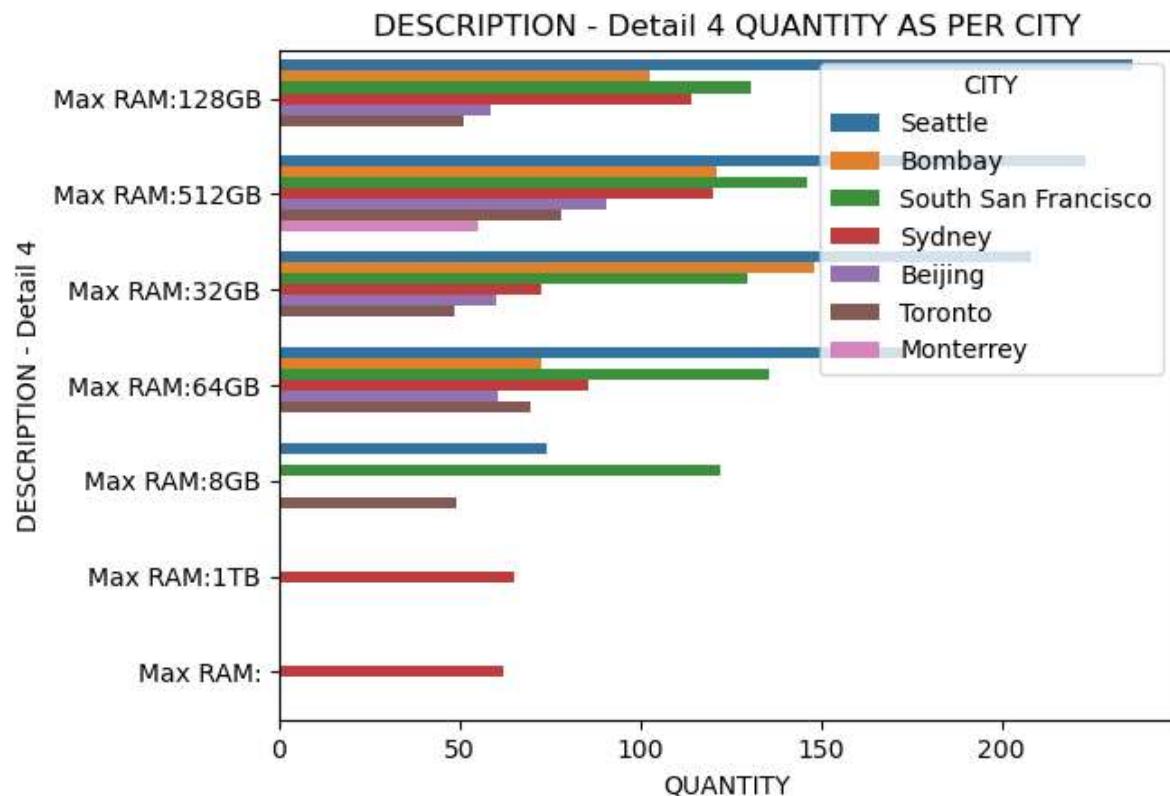


Observation:

- The above graph shows us the List_price of different MAX RAM as per State.
- From the above graph we can see that MAX RAM:8GB is very highly priced.
- We can also see that MAX RAM:8GB is very expensive in state like Washington and Maharashtra.
- MAX RAM:64GB is highly priced in Washington.
- MAX RAM:512&128GB is equally priced in Washington and Maharashtra.
- MAX RAM:32GB is also equally priced in Washington and Maharashtra.
- MAX RAM:1TB is equally priced in Maharashtra, Shenzhen and New South Wales.

42.DESCRIPTION - Detail 4 QUANTITY AS PER CITY

```
In [48]: A=Data.groupby(["DESCRIPTION - Detail 4","CITY"], as_index=False)[['QUANTITY']].sns.barplot(data=A,y="DESCRIPTION - Detail 4",x="QUANTITY",hue="CITY")plt.title("DESCRIPTION - Detail 4 QUANTITY AS PER CITY")plt.show()
```

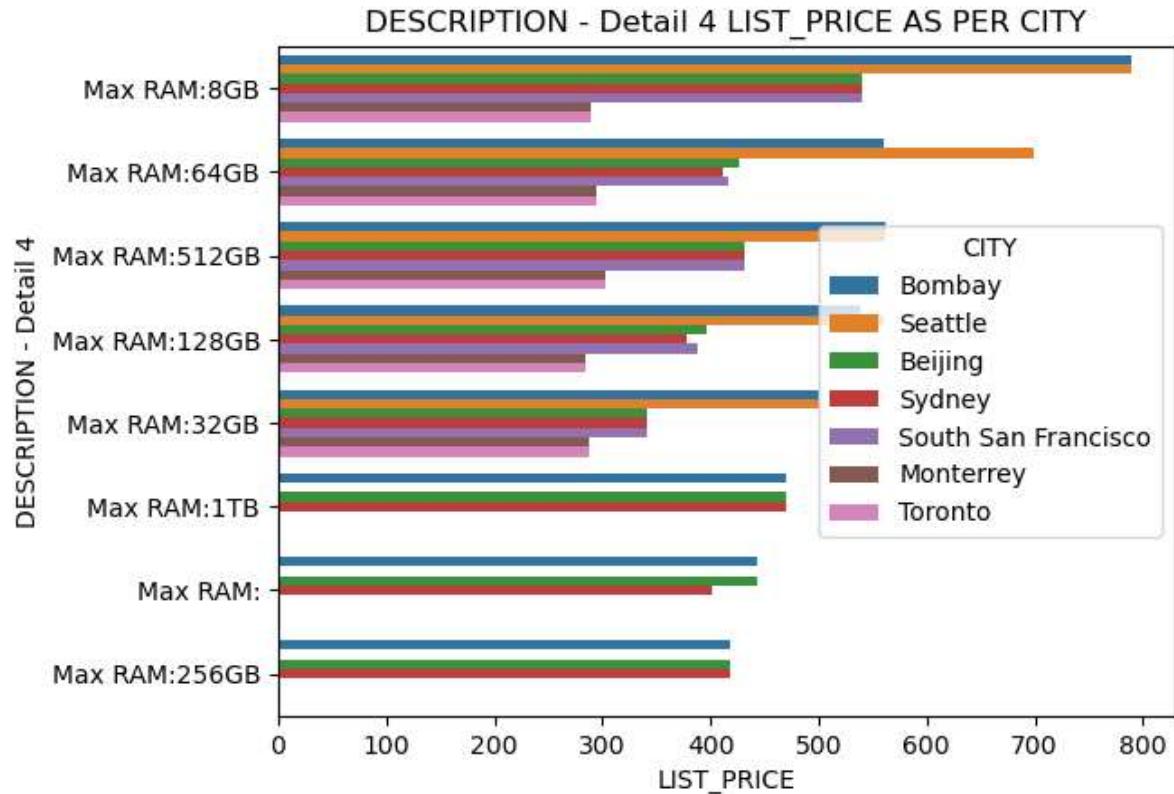


Observation:

- The above graph shows us the Quantity of different MAX RAM as per City.
- From the above graph we can see that MAX RAM:128GB has the highest number of quantity.
- In the above graph we can say that Seattle holds the maximum quantity of MAX RAM:128GB processor.
- From the above graph we can also see that Seattle hold the maximum quantity for all the MAX RAM except MAX RAM: and MAX RAM:1TB.

43.DESCRIPTION - Detail 4 LIST_PRICE AS PER CITY

```
In [42]: A=Data.groupby(["DESCRIPTION - Detail 4","CITY"], as_index=False)[['LIST_PRICE']]  
sns.barplot(data=A,y="DESCRIPTION - Detail 4",x="LIST_PRICE",hue="CITY")  
plt.title("DESCRIPTION - Detail 4 LIST_PRICE AS PER CITY")  
plt.show()
```



Observation:

- The above graph shows us the List_price of different MAX RAM as per City.
- From the above graph we can see that MAX RAM:8GB is very highly priced.
- We can also see that MAX RAM:8GB is very expensive in cities like Bombay and Seattle.
- We can also see that MAX RAM:64GB is highly priced in Seattle city.
- From the above graph we can see that MAX RAM:8GB and MAX RAM:64GB is equally priced in the cities like Monterrey and Toronto.
- We can see that Seattle city is not present in MAX RAM:,MAXRAM:1TB and MAX RAM:256GB prices.

```
In [ ]:
```

CONCLUSIN AND RECOMMENDATION

Hardware can contribute to cost reduction and decision making in various ways by enhancing efficiency improving productivity and enabling data driven insights such as:

- Automation and Efficiency:- Hardware technologies such as robotics, industrial automation systems and IoT(Internet of Things) devices that can automate manual tasks, streamline process and optimize resources utilization.
 - Decision Support System: Hardware-based decision support system integrate hardware components such as sensors, displays, CPU and user interfaces with software algorithms to provide real-time insights and support decision capabilities. These system enable users to visualize data, trends and simulate scenarios, facilitating informed decision making, across various domains such as Finance, Healthcare and supply-chain Management.
- Implications of Hardware in Inventory Management System.
- Implication of hardware in inventory management is significant as hardware technologies provide the necessary infrastructure for capturing, processing and analyzing inventory data efficiency. One such impact is:
 - Warehouse Management System:- Warehouse Management System are software applications that rely on hardware infrastructure such as servers, network, devices and database which can help us to manage and control warehouse operation.

In []: