Supply Chain Analysis using Python

- The supply Chain is the network of production and logistics involved in producing and delivering goods to customers. And Supply Chain Analysis means analyzing various components of a Supply Chain to understand how to improve the effectiveness of the Supply Chain to create more value for customers.

In [14]:

```
import pandas as pd
import numpy as np
import plotly.io as pio
import plotly.express as px
import plotly.graph_objects as go
import seaborn as sns
pio.templates.default = "plotly_white"
```

In [3]:

```
data = pd.read_csv("supply_chain_data.csv")
data.head()
```

Out[3]:

Customer demographics	Stock levels	Lead times	Order quantities	...	Location	Lead time	Production volumes	Manufacturing lead time	Manufacturing costs	Inspe re
Non-binary	58	7	96	...	Mumbai	29	215	29	46.279879	Pe
Female	53	30	37	...	Mumbai	23	517	30	33.616769	Pe
Unknown	1	10	88	...	Mumbai	12	971	27	30.688019	Pe
Non-binary	23	13	59	...	Kolkata	24	937	18	35.624741	
Non-binary	5	3	56	...	Delhi	5	414	3	92.065161	

In [4]:

```
1 data.describe()
```

Out[4]:

	Price	Availability	Number of products sold	Revenue generated	Stock levels	Lead times	Order quantities	Shipping times	10
count	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	
mean	49.462461	48.400000	460.990000	5776.048187	47.770000	15.960000	49.220000	5.750000	
std	31.168193	30.743317	303.780074	2732.841744	31.369372	8.785801	26.784429	2.724283	
min	1.699976	1.000000	8.000000	1061.618523	0.000000	1.000000	1.000000	1.000000	
25%	19.597823	22.750000	184.250000	2812.847151	16.750000	8.000000	26.000000	3.750000	
50%	51.239831	43.500000	392.500000	6006.352023	47.500000	17.000000	52.000000	6.000000	
75%	77.198228	75.000000	704.250000	8253.976921	73.000000	24.000000	71.250000	8.000000	
max	99.171329	100.000000	996.000000	9866.465458	100.000000	30.000000	96.000000	10.000000	

In [5]:

```
1 data.columns
```

Out[5]:

```
Index(['Product type', 'SKU', 'Price', 'Availability',
       'Number of products sold', 'Revenue generated', 'Customer demographics',
       'Stock levels', 'Lead times', 'Order quantities', 'Shipping times',
       'Shipping carriers', 'Shipping costs', 'Supplier name', 'Location',
       'Lead time', 'Production volumes', 'Manufacturing lead time',
       'Manufacturing costs', 'Inspection results', 'Defect rates',
       'Transportation modes', 'Routes', 'Costs'],
      dtype='object')
```

In [6]:

```
1 data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100 entries, 0 to 99
Data columns (total 24 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   Product type             100 non-null    object 
 1   SKU                      100 non-null    object 
 2   Price                    100 non-null    float64
 3   Availability             100 non-null    int64  
 4   Number of products sold  100 non-null    int64  
 5   Revenue generated        100 non-null    float64
 6   Customer demographics    100 non-null    object 
 7   Stock levels             100 non-null    int64  
 8   Lead times               100 non-null    int64  
 9   Order quantities         100 non-null    int64  
 10  Shipping times           100 non-null    int64  
 11  Shipping carriers        100 non-null    object 
 12  Shipping costs           100 non-null    float64
 13  Supplier name            100 non-null    object 
 14  Location                 100 non-null    object 
 15  Lead time                100 non-null    int64  
 16  Production volumes       100 non-null    int64  
 17  Manufacturing lead time  100 non-null    int64  
 18  Manufacturing costs      100 non-null    float64
 19  Inspection results       100 non-null    object 
 20  Defect rates             100 non-null    float64
 21  Transportation modes     100 non-null    object 
 22  Routes                   100 non-null    object 
 23  Costs                    100 non-null    float64
dtypes: float64(6), int64(9), object(9)
memory usage: 18.9+ KB
```

RELATIONSHIP BETWEEN PRICE OF THE PRODUCT AND REVENUE GENERATED BY THEM.

In [8]:

```
1 fig = px.scatter(data, x="Price", y="Revenue generated", color="Product type", hover_data=['Numb
2                  trendline="ols")
3 fig.show()
```

...

- Thus, the company derives more revenue from skincare products, and the higher the price of skincare products, the more revenue they generate.

SALES BY PRODUCT TYPE

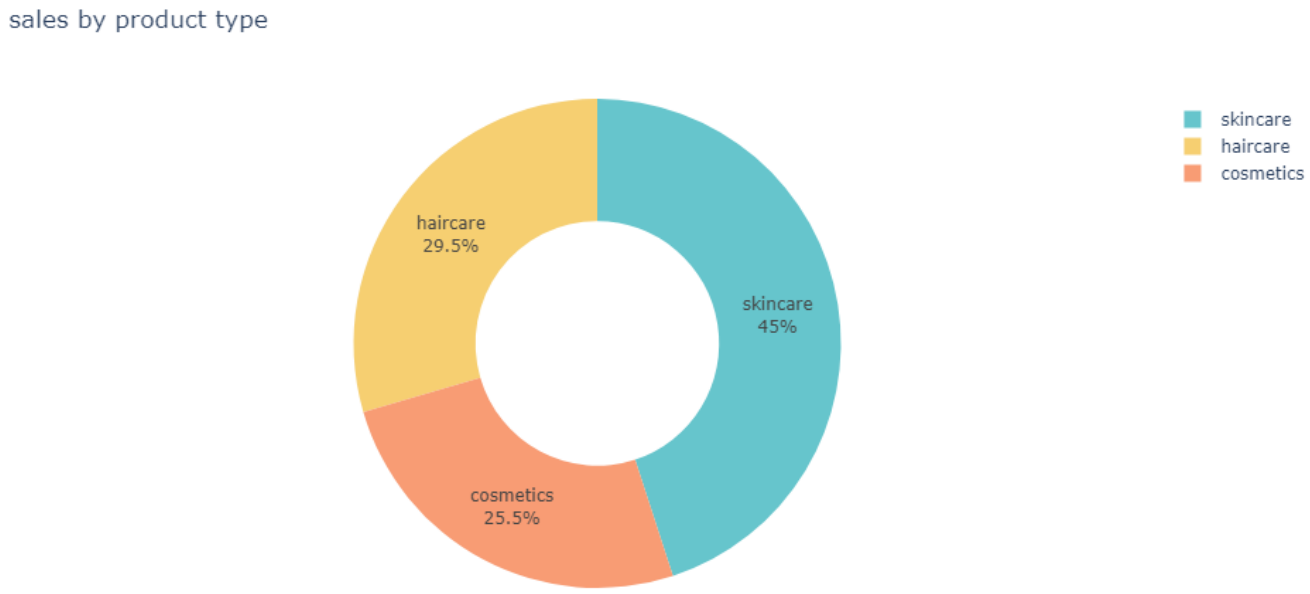
In [10]:

```
sales_by_product = data.groupby("Product type")["Number of products sold"].sum().reset_index()
```

In [11]:

```
fig = px.pie(sales_by_product,values = "Number of products sold", names="Product type",
             title = "sales by product type", hover_data=['Number of products sold'],
                    hole=0.5,
                    color_discrete_sequence=px.colors.qualitative.Pastel)
fig.update_traces(textposition='inside', textinfo='percent+label')
fig.show()
```

. . .



- So 45% of the business comes from skincare products, 29.5% from haircare, and 25.5% from cosmetics.

TOTAL REVENUE GENERATED FROM SHIPPING CARRIERS.

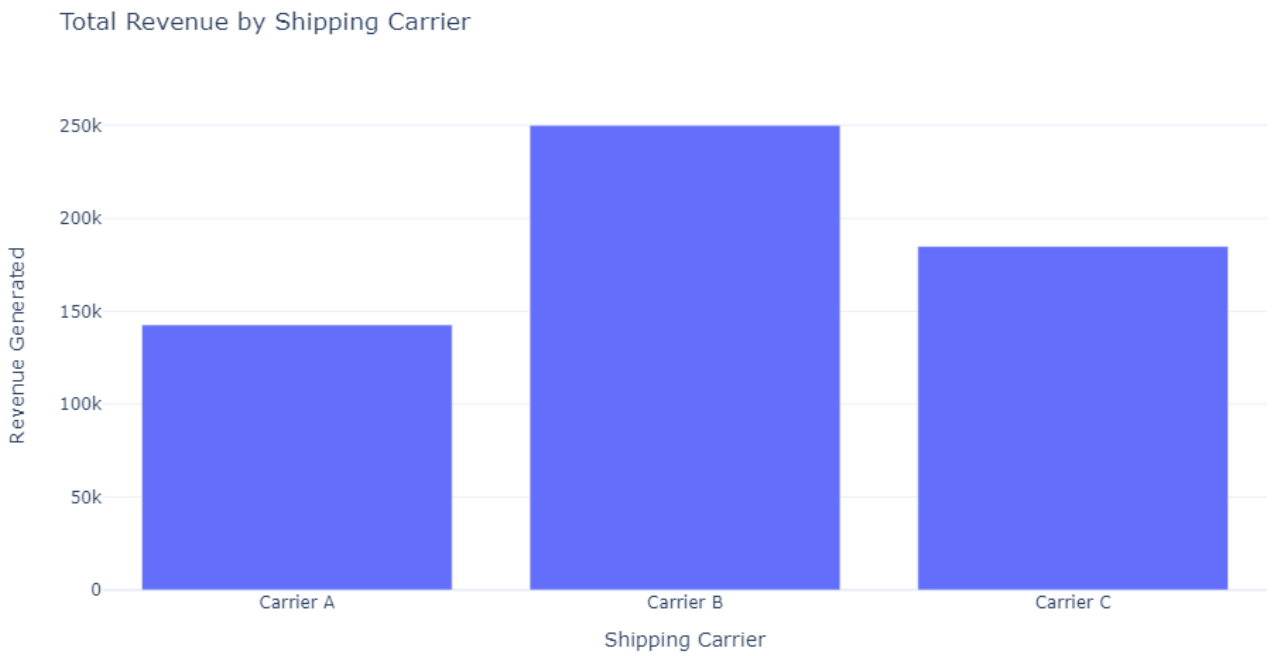
In [13]:

```
total_revenue = data.groupby("Shipping carriers")["Revenue generated"].sum().reset_index()
```

In [18]:

```
total_revenue = data.groupby('Shipping carriers')['Revenue generated'].sum().reset_index()
fig = go.Figure()
fig.add_trace(go.Bar(x=total_revenue['Shipping carriers'],
                     y=total_revenue['Revenue generated']))
fig.update_layout(title='Total Revenue by Shipping Carrier',
                  xaxis_title='Shipping Carrier',
                  yaxis_title='Revenue Generated')
fig.show()
```

. . .



- So the company is using three carriers for transportation, and Carrier B helps the company in generating more revenue.

AVERAGE LEAD TIME PER PRODUCT AND AVERAGE MANUFACTURING COSTS FOR ALL PRODUCTS.

In [19]:

```
avg_lead_time = data.groupby("Product type")["Lead times"].mean().reset_index()
avg_mfg_cost = data.groupby("Product type")["Manufacturing costs"].mean().reset_index()
```

In [20]:

```
res = pd.merge(avg_lead_time, avg_mfg_cost, on="Product type")
res.rename(columns = {"Lead times":"avg lead times", "Manufacturing costs":"avg mfg costs"})
res.head()
```

Out[20]:

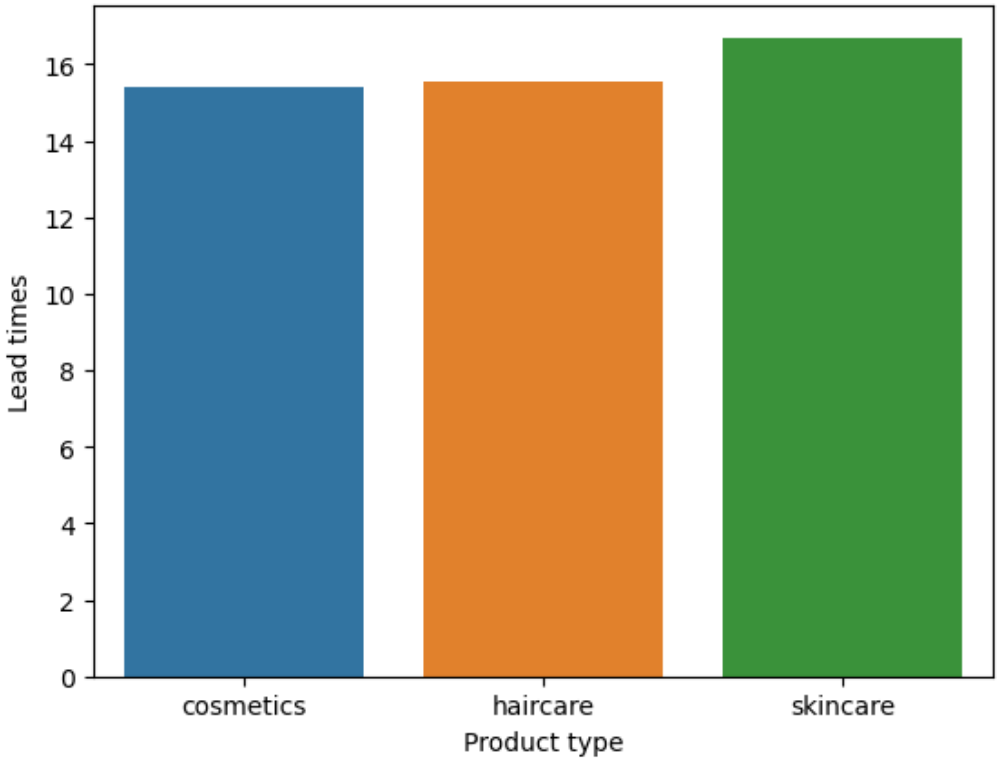
	Product type	Lead times	Manufacturing costs
0	cosmetics	15.384615	43.052740
1	haircare	15.529412	48.457993
2	skincare	16.700000	48.993157

In [21]:

```
1 sns.barplot(res, x="Product type", y="Lead times")
```

Out[21]:

```
<Axes: xlabel='Product type', ylabel='Lead times'>
```

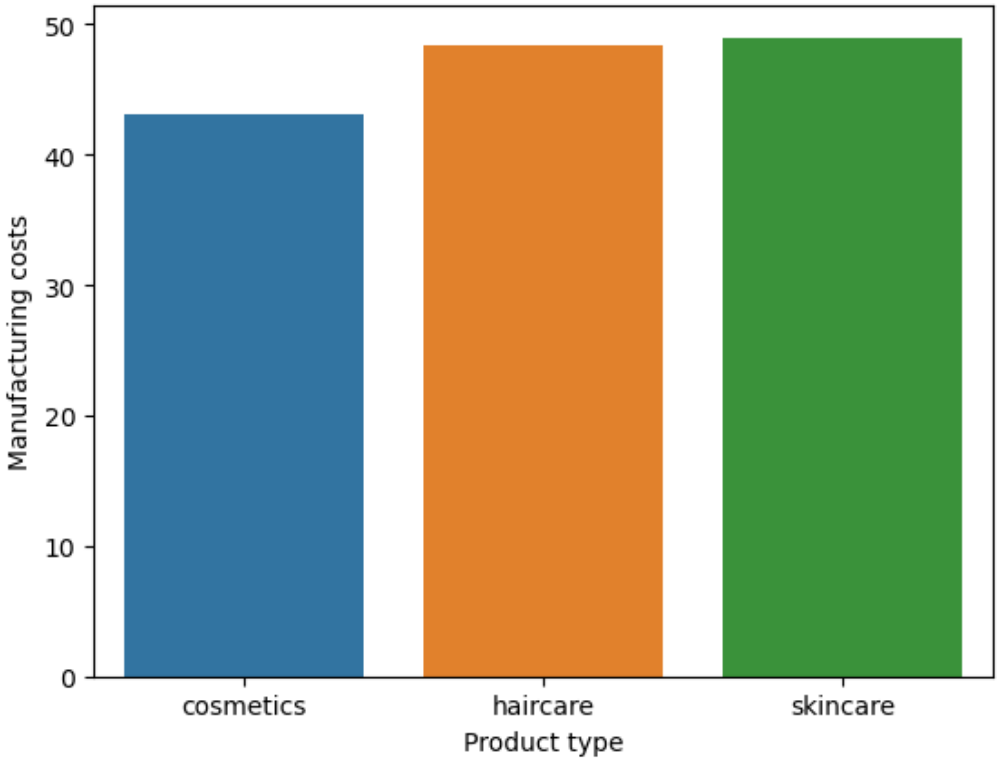


In [22]:

```
1 sns.barplot(res, x="Product type", y="Manufacturing costs")
```

Out[22]:

```
<Axes: xlabel='Product type', ylabel='Manufacturing costs'>
```



- There is no significant diffrence between the lead times of all products or manufacturning costs of all products.

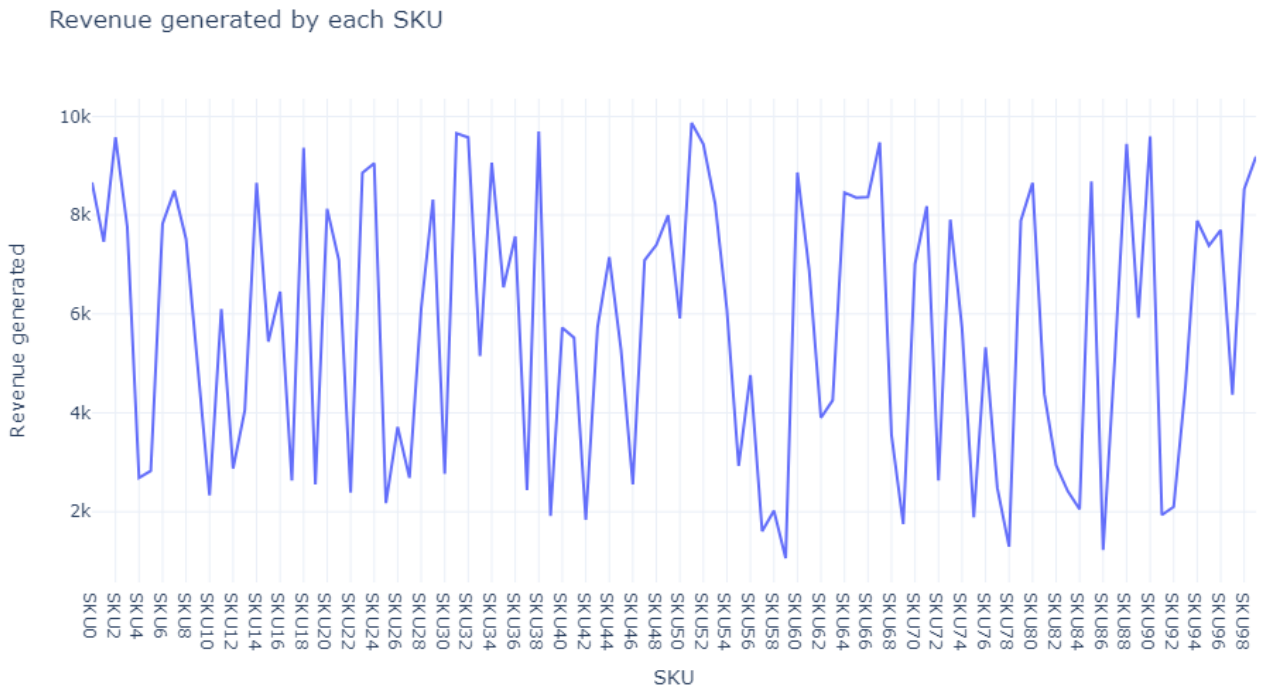
ANALYSING SKU'S

- SKU stands for Stock Keeping Units. They’re like special codes that help companies keep track of all the different things they have for sale.
- Imagine we have a large toy store with lots of toys. Each toy is different and has its name and price, but when you want to know how many you have left, you need a way to identify them. So you give each toy a unique code, like a secret number only the store knows. This secret number is called SKU.

REVENUE GENERATED BY EACH SKU

In [24]:

```
1 fig = px.line(data,x = "SKU", y = "Revenue generated", title = "Revenue generated by each SKU")
2 fig.show()
```



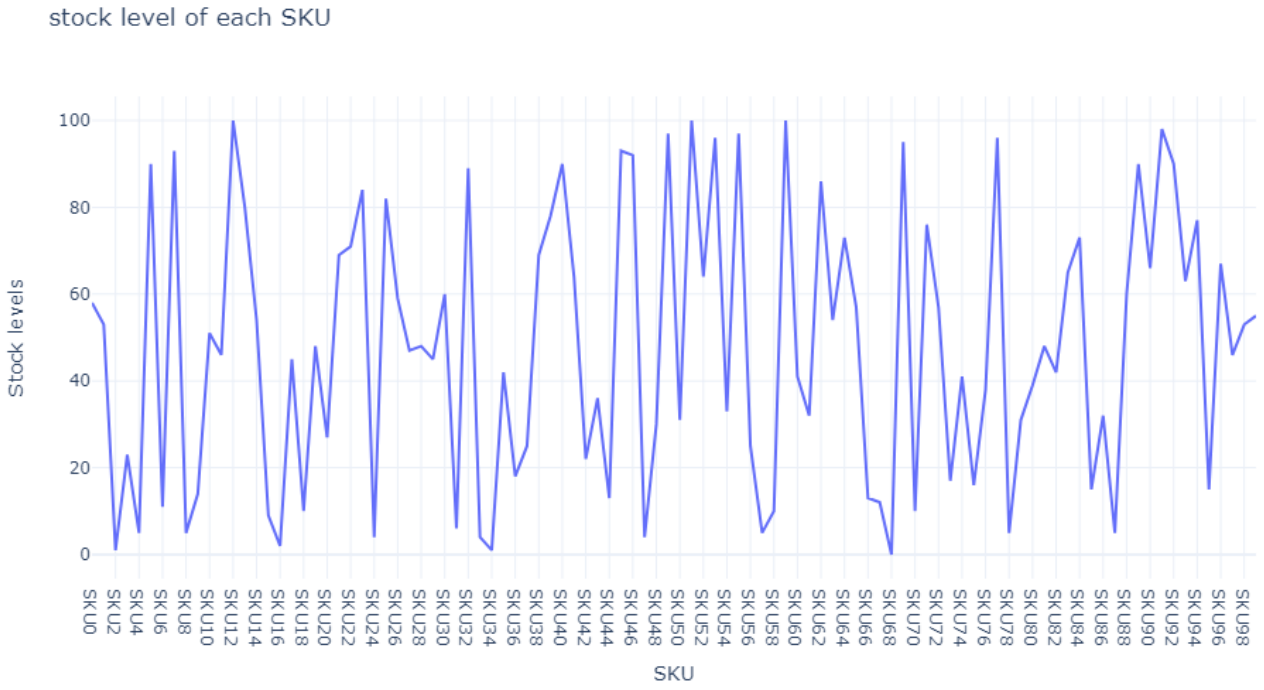
STOCK LEVEL OF EACH SKU

- Stock levels refer to the number of products a store or business has in its inventory.

In [27]:

```
1 fig = px.line(data,x="SKU",y="Stock levels",title = "stock level of each SKU")
2 fig.show()
```

...

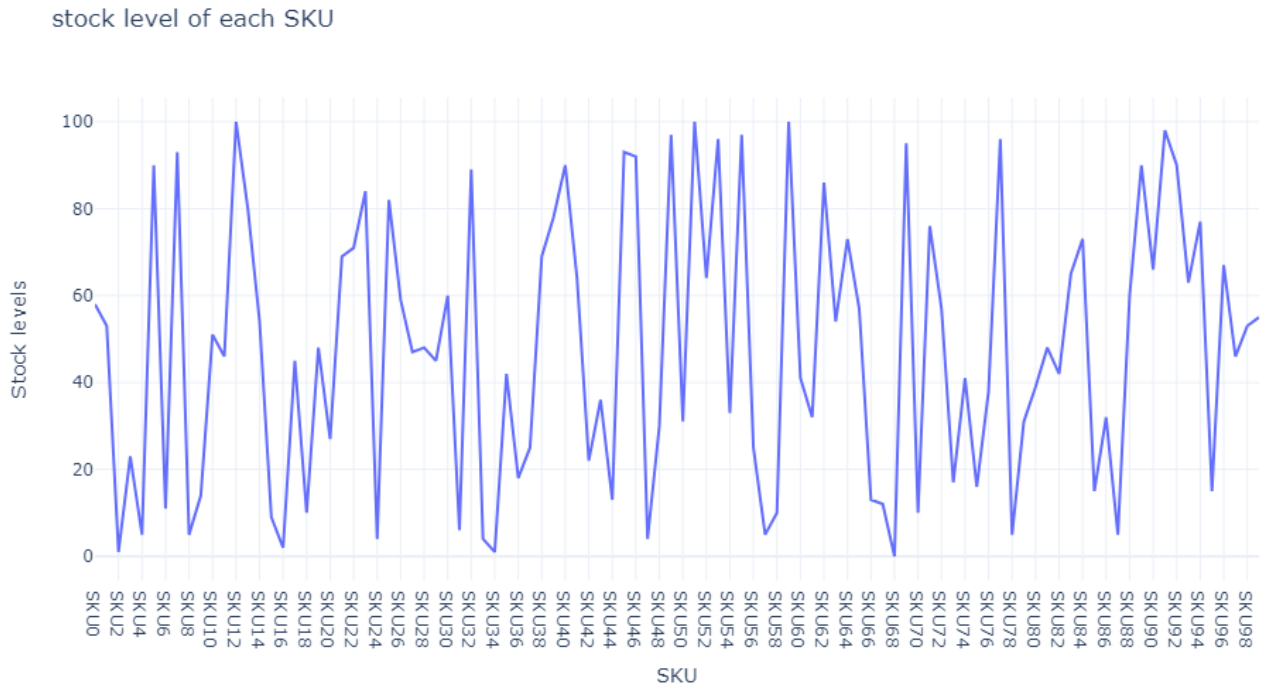


ORDER QUANTITY OF EACH SKU

In [29]:

```
fig = px.bar(data, x="SKU", y="Order quantities", title = "order quantity of each SKU")
fig.show()
```

...



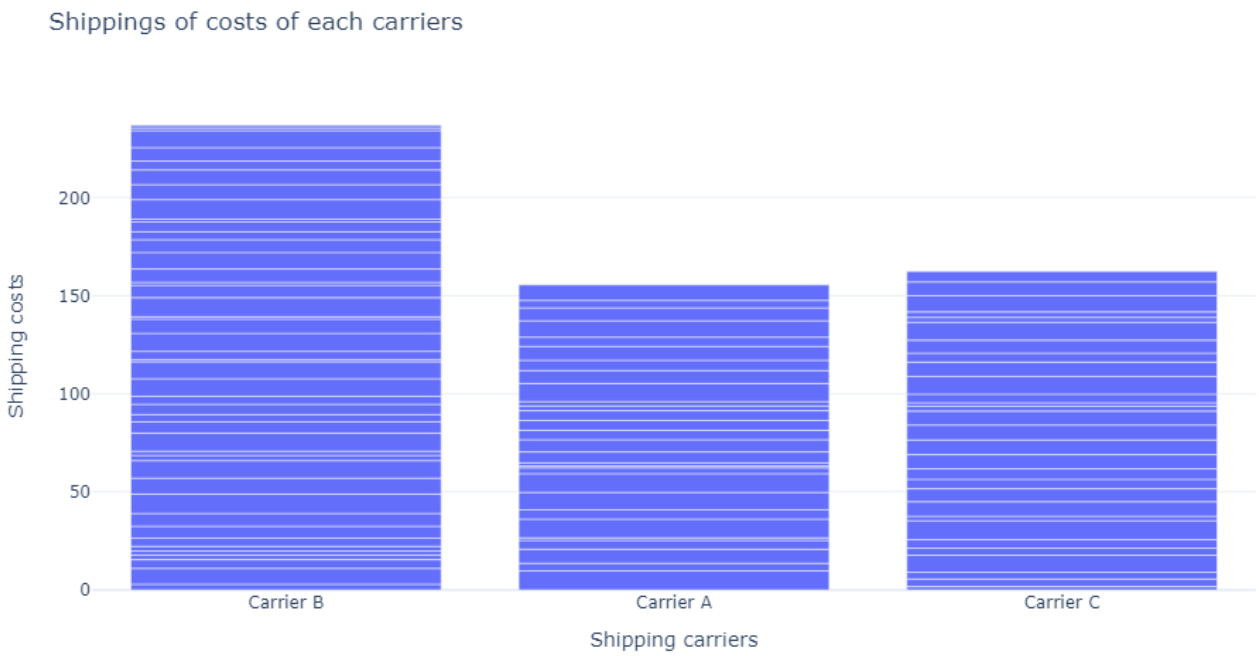
COST ANALYSIS

SHIPPING COSTS OF CARRIERS

In [31]:

```
fig = px.bar(data,x="Shipping carriers", y="Shipping costs", title="Shippings of costs of each c
fig.show()
```

...



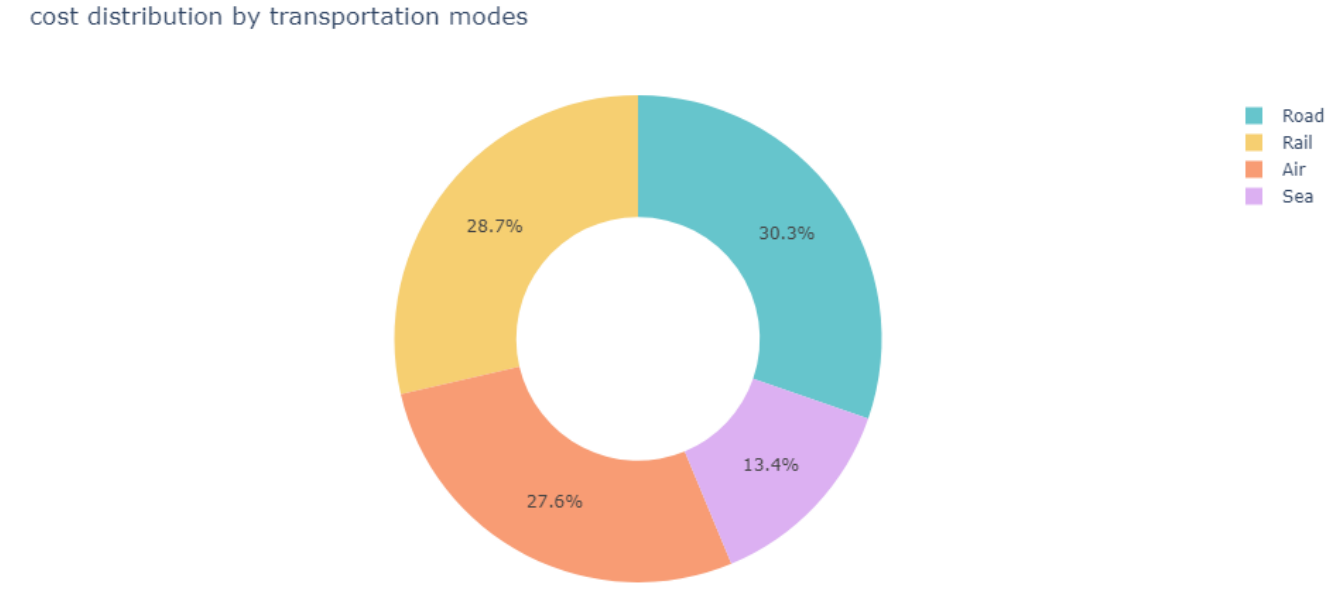
- we discovered that Carrier B helps the company in more revenue. It is also the most costly Carrier among the three.

COST DISTRIBUTION BY TRANSPORTATION MODES

In [36]:

```
fig = px.pie(data,values="Costs",names="Transportation modes",title="cost distribution by transp
             hole=0.5,
             color_discrete_sequence=px.colors.qualitative.Pastel)
fig.show()
```

...



ANALYSING DEFECT RATE.

AVERAGE DEFECT RATE PER PRODUCT

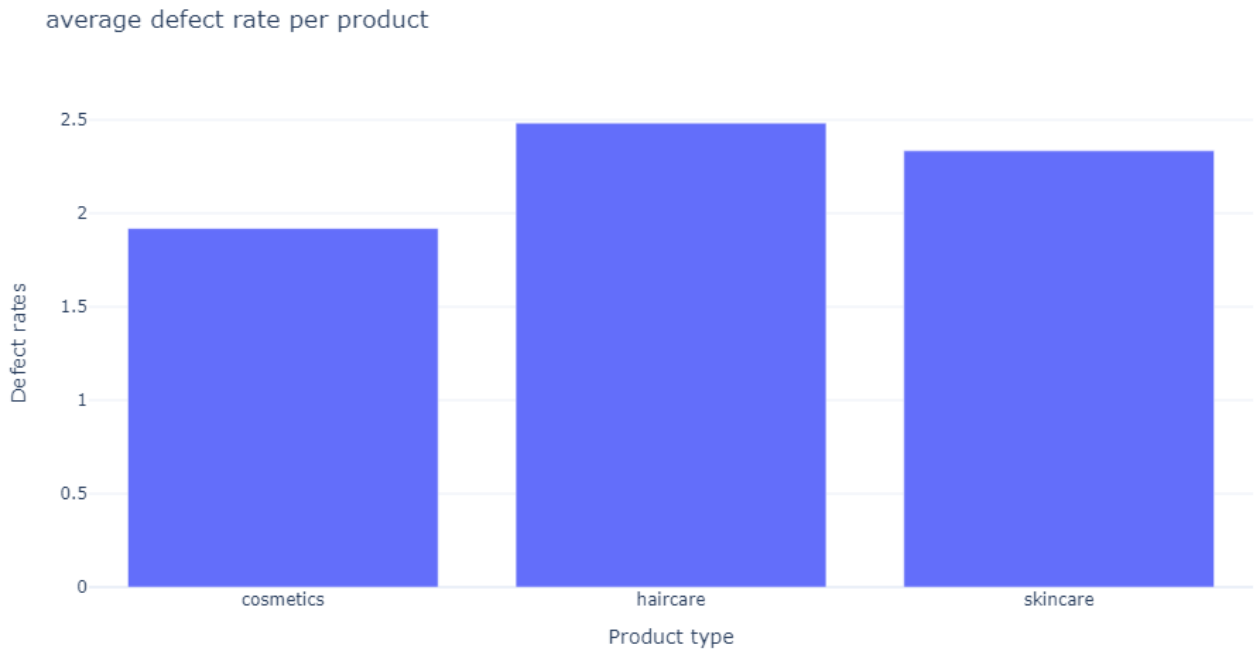
In [37]:

```
avg_defect_rate = data.groupby("Product type")["Defect rates"].mean().reset_index()
```

In [38]:

```
fig = px.bar(avg_defect_rate,x="Product type",y="Defect rates",title="average defect rate per pr
fig.show()
```

...



- So the defect rate of haircare products is higher.

DEFECT RATES BY TRANSPORTATION MODES

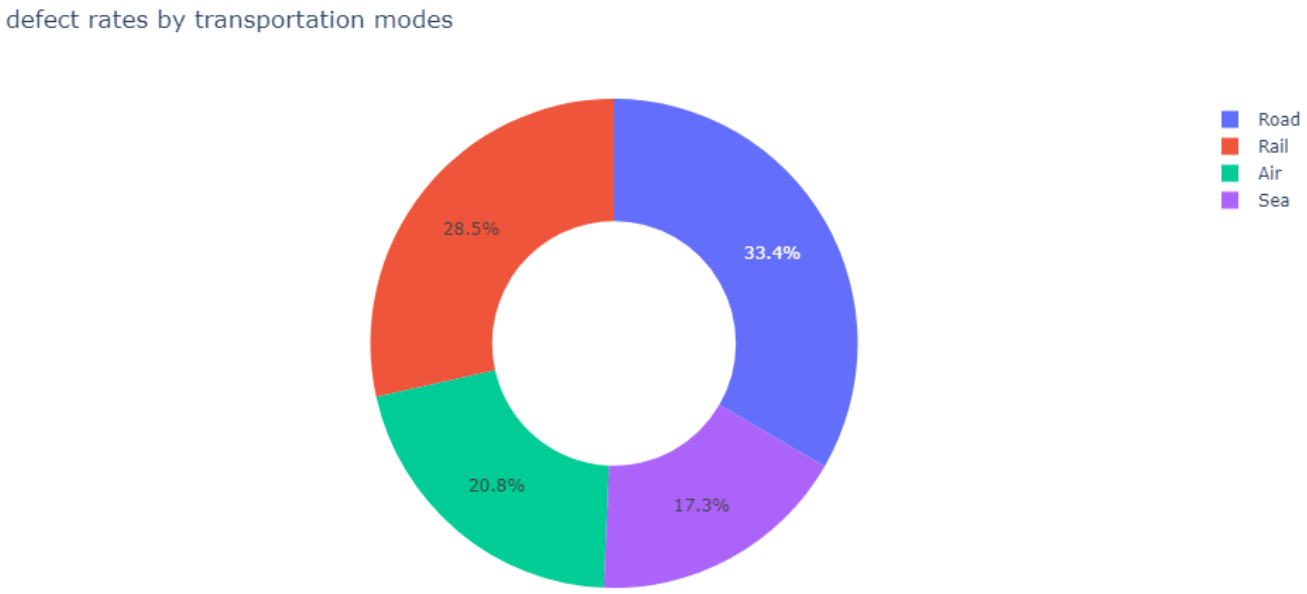
In [42]:

```
pivot = pd.pivot_table(data, values="Defect rates",index=["Transportation modes"],aggfunc='mean'
```

In [47]:

```
fig = px.pie(data, values="Defect rates", names="Transportation modes",title="defect rates by tr
             hole=0.5)
fig.show()
```

...



- Road transportation results in a higher defect rate, and Air transportation has the lowest defect rate.

- Supply Chain Analysis means analyzing various components of a Supply Chain to understand how to improve the effectiveness of the Supply Chain to create more value for customers.