Supply Chain Analysis

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.

Step 1: Import Libraries

In [1]:

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

In [2]:

```
data = pd.read_csv('supply_chain_data.csv')
```

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

Out[3]:

	Product type	SKU	Price	Availability	Number of products sold	Revenue generated	Customer demographics	Stock levels	Lead times	Order quantities	...	Location	Lead time	Producti volum
0	haircare	SKU0	69.808006	55	802	8661.996792	Non-binary	58	7	96	...	Mumbai	29	2
1	skincare	SKU1	14.843523	95	736	7460.900065	Female	53	30	37	...	Mumbai	23	5
2	haircare	SKU2	11.319683	34	8	9577.749626	Unknown	1	10	88	...	Mumbai	12	9
3	skincare	SKU3	61.163343	68	83	7766.836426	Non-binary	23	13	59	...	Kolkata	24	9.
4	skincare	SKU4	4.805496	26	871	2686.505152	Non-binary	5	3	56	...	Delhi	5	4

5 rows × 24 columns

Step 2: Find null values

In [4]:

```python
data.isnull()
```

Out[4]:

	Product type	SKU	Price	Availability	Number of products sold	Revenue generated	Customer demographics	Stock levels	Lead times	Order quantities	...	Location	Lead time	Production volumes	M
0	False	False	False	False	False	False	False	False	False	False	...	False	False	False	
1	False	False	False	False	False	False	False	False	False	False	...	False	False	False	
2	False	False	False	False	False	False	False	False	False	False	...	False	False	False	
3	False	False	False	False	False	False	False	False	False	False	...	False	False	False	
4	False	False	False	False	False	False	False	False	False	False	...	False	False	False	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
95	False	False	False	False	False	False	False	False	False	False	...	False	False	False	
96	False	False	False	False	False	False	False	False	False	False	...	False	False	False	
97	False	False	False	False	False	False	False	False	False	False	...	False	False	False	
98	False	False	False	False	False	False	False	False	False	False	...	False	False	False	
99	False	False	False	False	False	False	False	False	False	False	...	False	False	False	

100 rows × 24 columns

```
In [5]: data.isnull().sum()
```

```
Out[5]: Product type               0
        SKU                        0
        Price                      0
        Availability               0
        Number of products sold    0
        Revenue generated          0
        Customer demographics      0
        Stock levels               0
        Lead times                 0
        Order quantities           0
        Shipping times             0
        Shipping carriers          0
        Shipping costs             0
        Supplier name              0
        Location                   0
        Lead time                  0
        Production volumes         0
        Manufacturing lead time    0
        Manufacturing costs        0
        Inspection results         0
        Defect rates               0
        Transportation modes       0
        Routes                     0
        Costs                      0
        dtype: int64
```

In [6]:

```
data.astype
```

```
Out[6]: <bound method NDFrame.astype of    Product type    SKU      Price  Availability  Number of products sold  \
        0      haircare   SKU0  69.808006            55                      802   
        1      skincare   SKU1  14.843523            95                      736   
        2      haircare   SKU2  11.319683            34                        8   
        3      skincare   SKU3  61.163343            68                       83   
        4      skincare   SKU4   4.805496            26                      871   
        ..          ...    ...        ...           ...                      ...   
        95     haircare  SKU95  77.903927            65                      672   
        96    cosmetics  SKU96  24.423131            29                      324   
        97     haircare  SKU97   3.526111            56                       62   
        98     skincare  SKU98  19.754605            43                      913   
        99     haircare  SKU99  68.517833            17                      627   

            Revenue generated Customer demographics  Stock levels  Lead times  \
        0         8661.996792            Non-binary            58           7   
        1         7460.900065                Female            53          30   
        2         9577.749626               Unknown             1          10   
        3         7766.836426            Non-binary            23          13   
        4         2686.505152            Non-binary             5           3   
        ..                ...                   ...           ...         ...   
        95        7386.363944               Unknown            15          14   
        96        7698.424766            Non-binary            67           2   
        97        4370.916580                  Male            46          19   
        98        8525.952560                Female            53           1   
        99        9185.185829               Unknown            55           8   

            Order quantities  ...  Location Lead time  Production volumes  \
        0                 96  ...    Mumbai        29                 215   
        1                 37  ...    Mumbai        23                 517   
        2                 88  ...    Mumbai        12                 971   
        3                 59  ...   Kolkata        24                 937   
        4                 56  ...     Delhi         5                 414   
        ..               ...  ...       ...       ...                 ...   
        95                26  ...    Mumbai        18                 450   
        96                32  ...    Mumbai        28                 648   
        97                 4  ...    Mumbai        10                 535   
        98                27  ...   Chennai        28                 581   
        99                59  ...   Chennai        29                 921   

           Manufacturing lead time Manufacturing costs  Inspection results  \
        0                       29           46.279879             Pending   
        1                       30           33.616769             Pending   
        2                       27           30.688019             Pending   
```

```
3                       18           35.624741                Fail
4                        3           92.065161                Fail
..                     ...                 ...                 ...
95                      26           58.890686             Pending
96                      28           17.803756             Pending
97                      13           65.765156                Fail
98                       9            5.604691             Pending
99                       2           38.072899                Fail

    Defect rates  Transportation modes   Routes       Costs
0       0.226410                  Road  Route B  187.752075
1       4.854068                  Road  Route B  503.065579
2       4.580593                   Air  Route C  141.920282
3       4.746649                  Rail  Route A  254.776159
4       3.145580                   Air  Route A  923.440632
..           ...                   ...      ...         ...
95      1.210882                   Air  Route A  778.864241
96      3.872048                  Road  Route A  188.742141
97      3.376238                  Road  Route A  540.132423
98      2.908122                  Rail  Route A  882.198864
99      0.346027                  Rail  Route B  210.743009

[100 rows x 24 columns]>
```

Step 3: Descriptive Statistics report of the Datasets

In [7]:

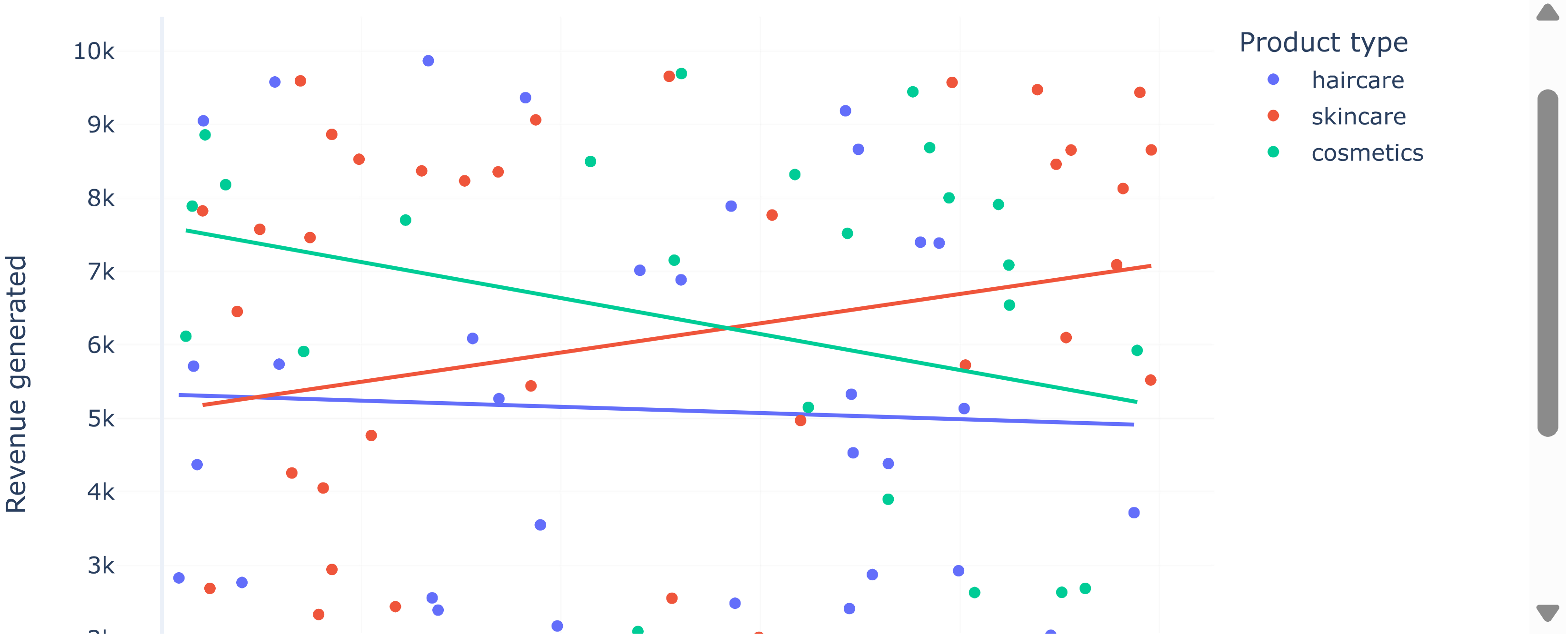
```
data.describe()
```

Out[7]:

	Price	Availability	Number of products sold	Revenue generated	Stock levels	Lead times	Order quantities	Shipping times	Shipping costs	Lead time	Produ vo
count	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.0
mean	49.462461	48.400000	460.990000	5776.048187	47.770000	15.960000	49.220000	5.750000	5.548149	17.080000	567.8
std	31.168193	30.743317	303.780074	2732.841744	31.369372	8.785801	26.784429	2.724283	2.651376	8.846251	263.0
min	1.699976	1.000000	8.000000	1061.618523	0.000000	1.000000	1.000000	1.000000	1.013487	1.000000	104.0
25%	19.597823	22.750000	184.250000	2812.847151	16.750000	8.000000	26.000000	3.750000	3.540248	10.000000	352.0
50%	51.239831	43.500000	392.500000	6006.352023	47.500000	17.000000	52.000000	6.000000	5.320534	18.000000	568.5
75%	77.198228	75.000000	704.250000	8253.976921	73.000000	24.000000	71.250000	8.000000	7.601695	25.000000	797.0
max	99.171329	100.000000	996.000000	9866.465458	100.000000	30.000000	96.000000	10.000000	9.929816	30.000000	985.0

Step 4: Analyzing the relationship between the price of the products and the revenue generated by using the Scatter plot:

```
In [9]: fig = px.scatter(data, x='Price', y='Revenue generated', color = 'Product type', hover_data= ['Number of prod
        fig.show()
```



Step 5: Sales by Product type

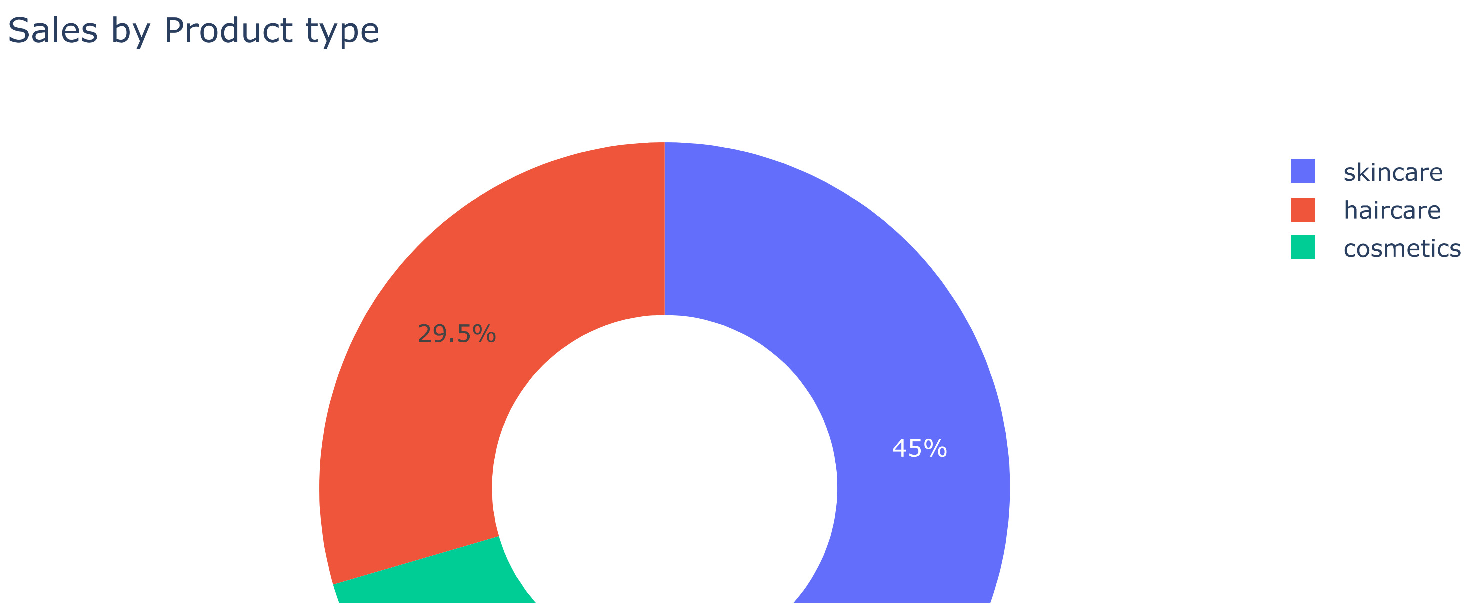
```
In [10]: sales_data = data.groupby('Product type')['Number of products sold'].sum().reset_index()
```

```
In [11]: sales_data
```

Out[11]:

	Product type	Number of products sold
0	cosmetics	11757
1	haircare	13611
2	skincare	20731

In [12]:
```
pie_chart = px.pie(sales_data, values = 'Number of products sold', names='Product type',
                   title='Sales by Product type', hover_data=['Number of products sold'], hole=0.5)
pie_chart.show()
```



Step 6: Revenue from Shipping carriers

In [13]:
```
total_revenue = data.groupby('Shipping carriers')['Revenue generated'].sum().reset_index()
```

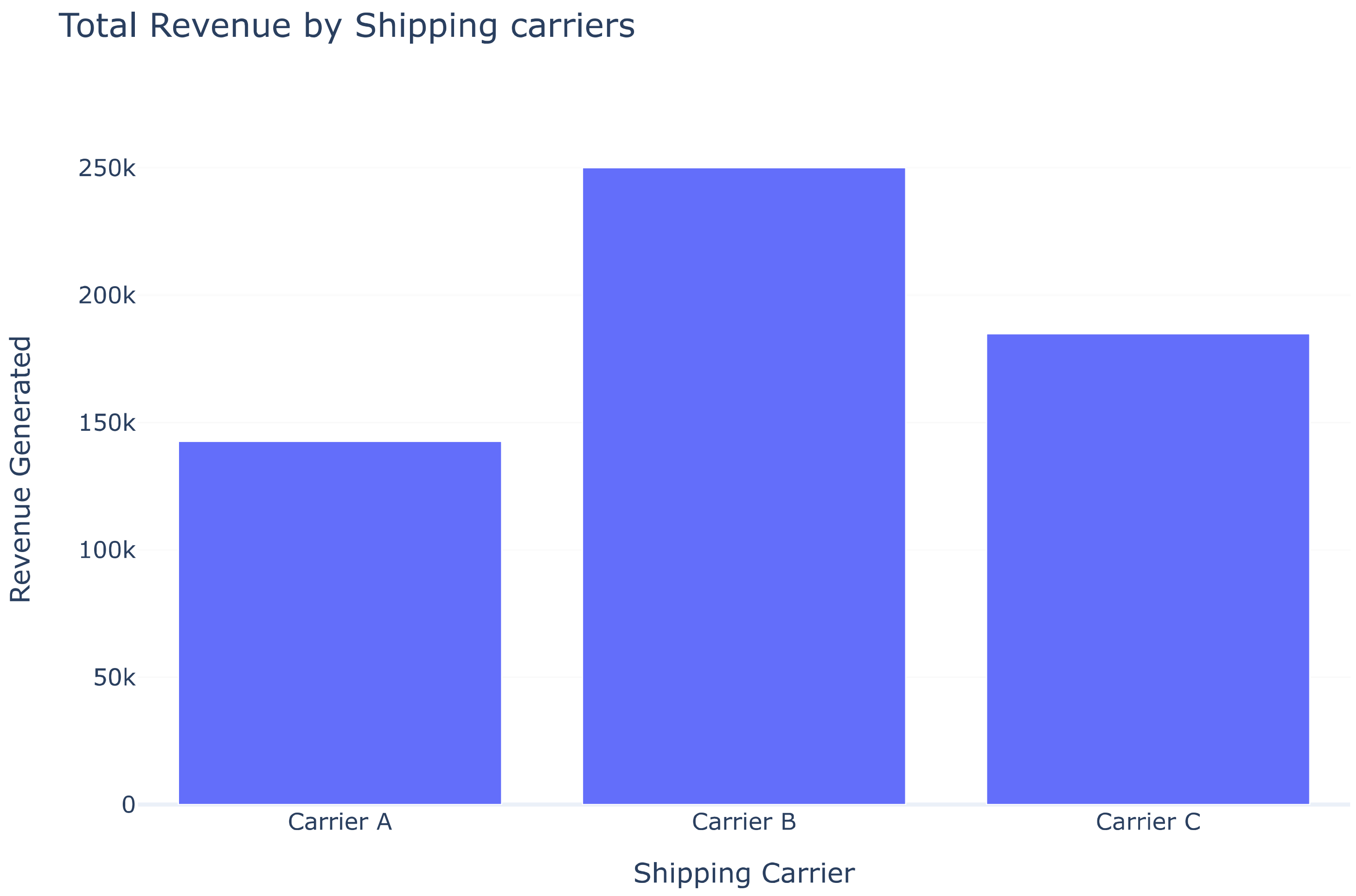
In [14]:
```
total_revenue
```

Out[14]:

	Shipping carriers	Revenue generated
0	Carrier A	142629.994607
1	Carrier B	250094.646988
2	Carrier C	184880.177143

In [15]:

```python
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 carriers',
                 xaxis_title = 'Shipping Carrier',
                 yaxis_title = 'Revenue Generated')
fig.show()
```



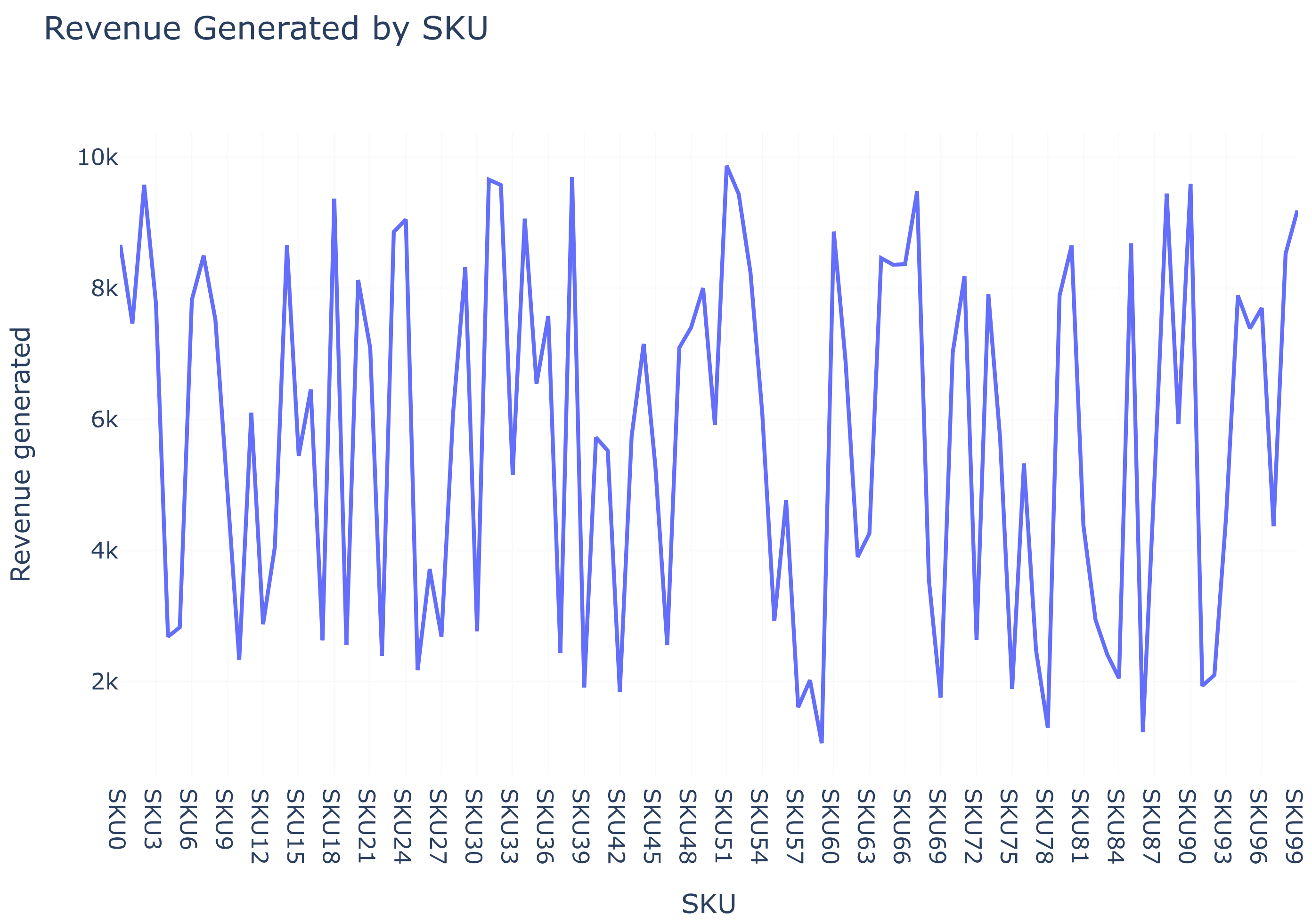
Step 7: Average lead time and Average Manufacturing Costs for all products of the company

```
In [16]: lead_time = data.groupby('Product type')['Lead time'].mean().reset_index()
         manufacturing_costs = data.groupby('Product type')['Manufacturing costs'].mean().reset_index()
         lt = pd.merge(avg_lead_time, avg_manufacturing_costs, on='Product type')
         lt.rename(columns={'Lead time': 'Average Lead Time', 'Manufacturing costs': 'Average Manufacturing costs'}, in
         t(avg_lead_time)
         t(avg_manufacturing_costs)
         t(result)
```

```
  Product type  Lead time
0    cosmetics  13.538462
1     haircare  18.705882
2     skincare  18.000000
  Product type  Manufacturing costs
0    cosmetics            43.052740
1     haircare            48.457993
2     skincare            48.993157
  Product type  Average Lead Time  Average Manufacturing costs
0    cosmetics          13.538462                    43.052740
1     haircare          18.705882                    48.457993
2     skincare          18.000000                    48.993157
```

SKU (Stock Keeping Units)

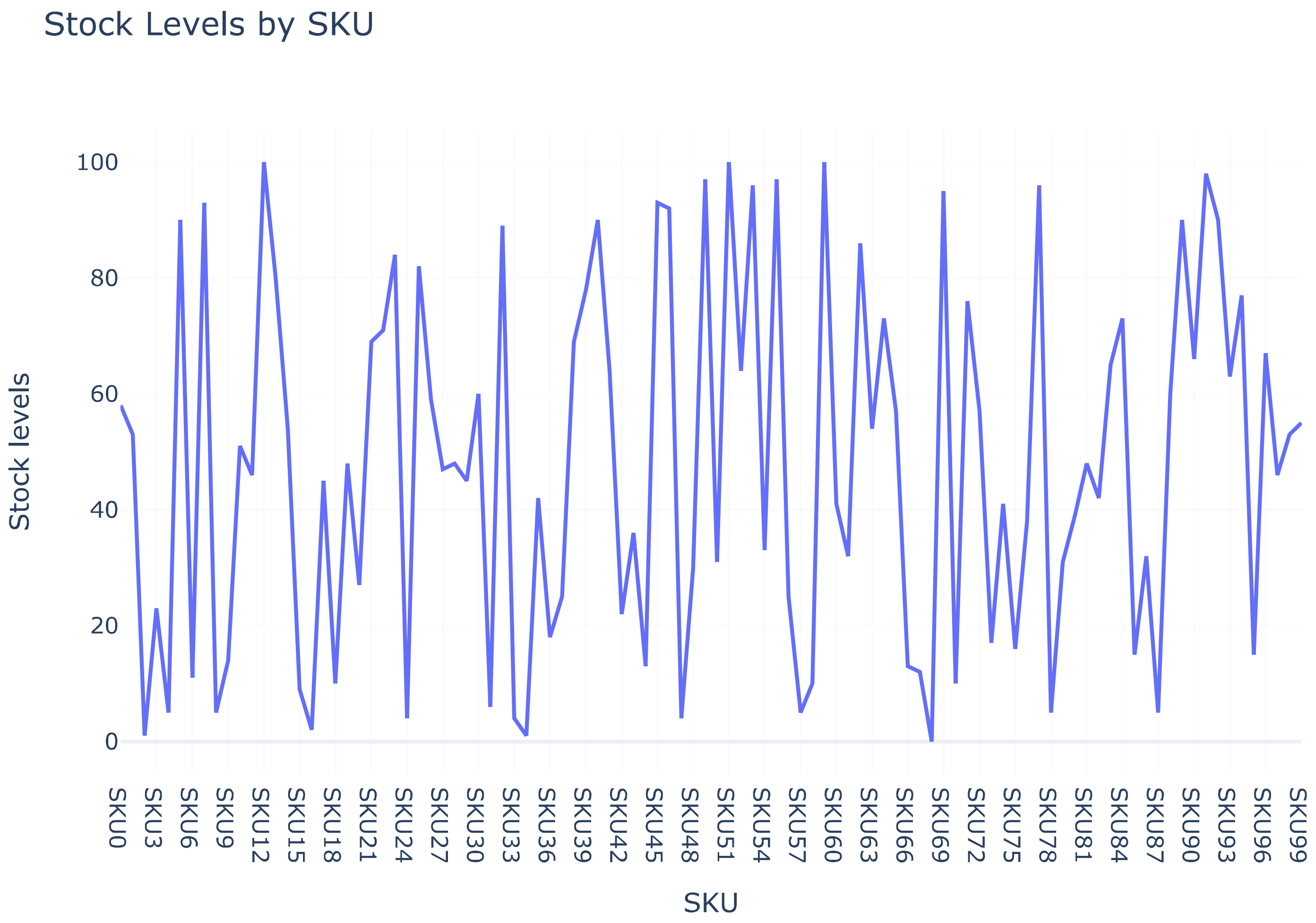
In [17]:
```
revenue_chart = px.line(data, x='SKU', y='Revenue generated',
                        title = 'Revenue Generated by SKU')
revenue_chart.show()
```



Stock Level Chart

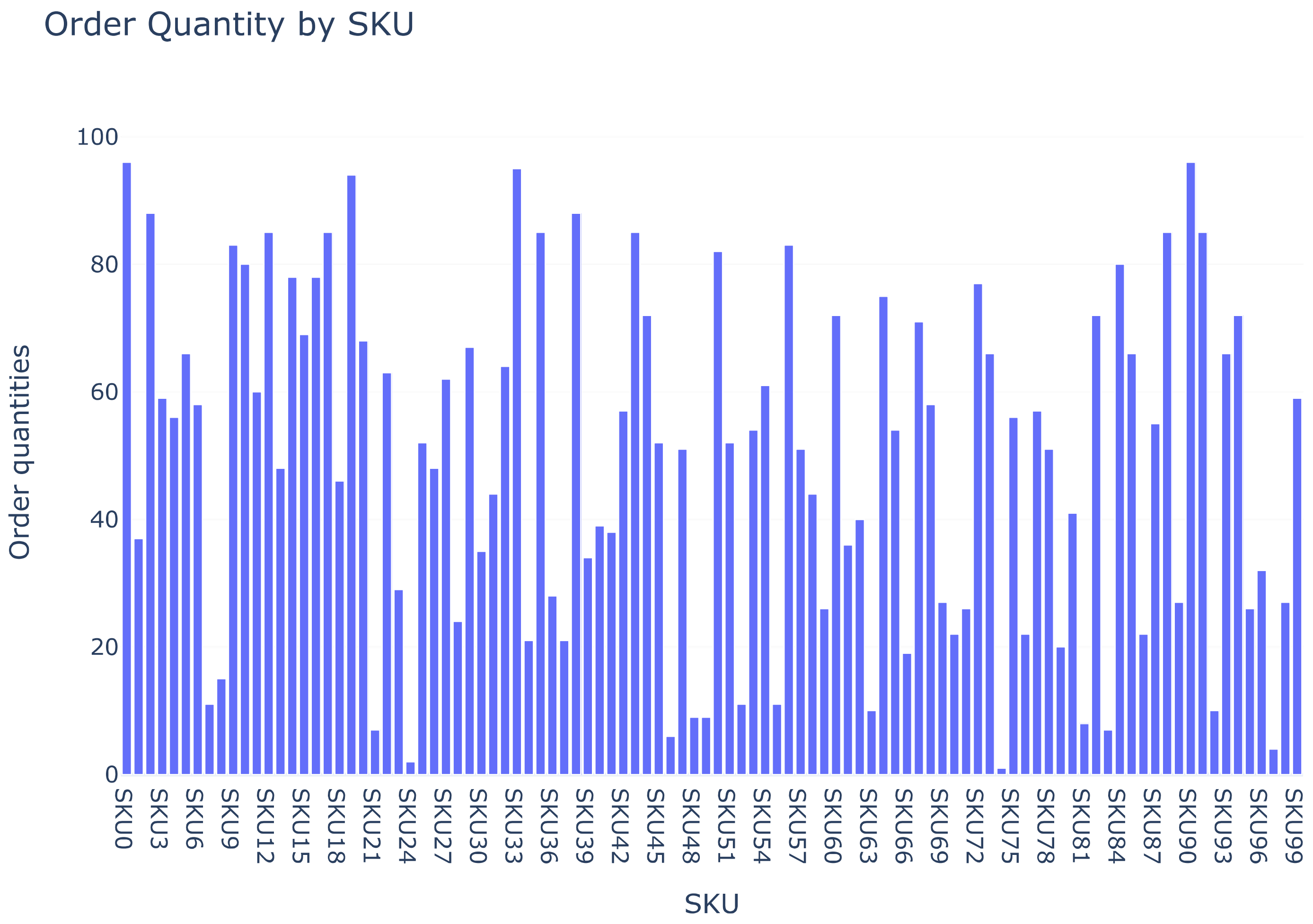
In [18]:

```
stock_chart = px.line(data, x='SKU', y='Stock levels',
                      title='Stock Levels by SKU')
stock_chart.show()
```



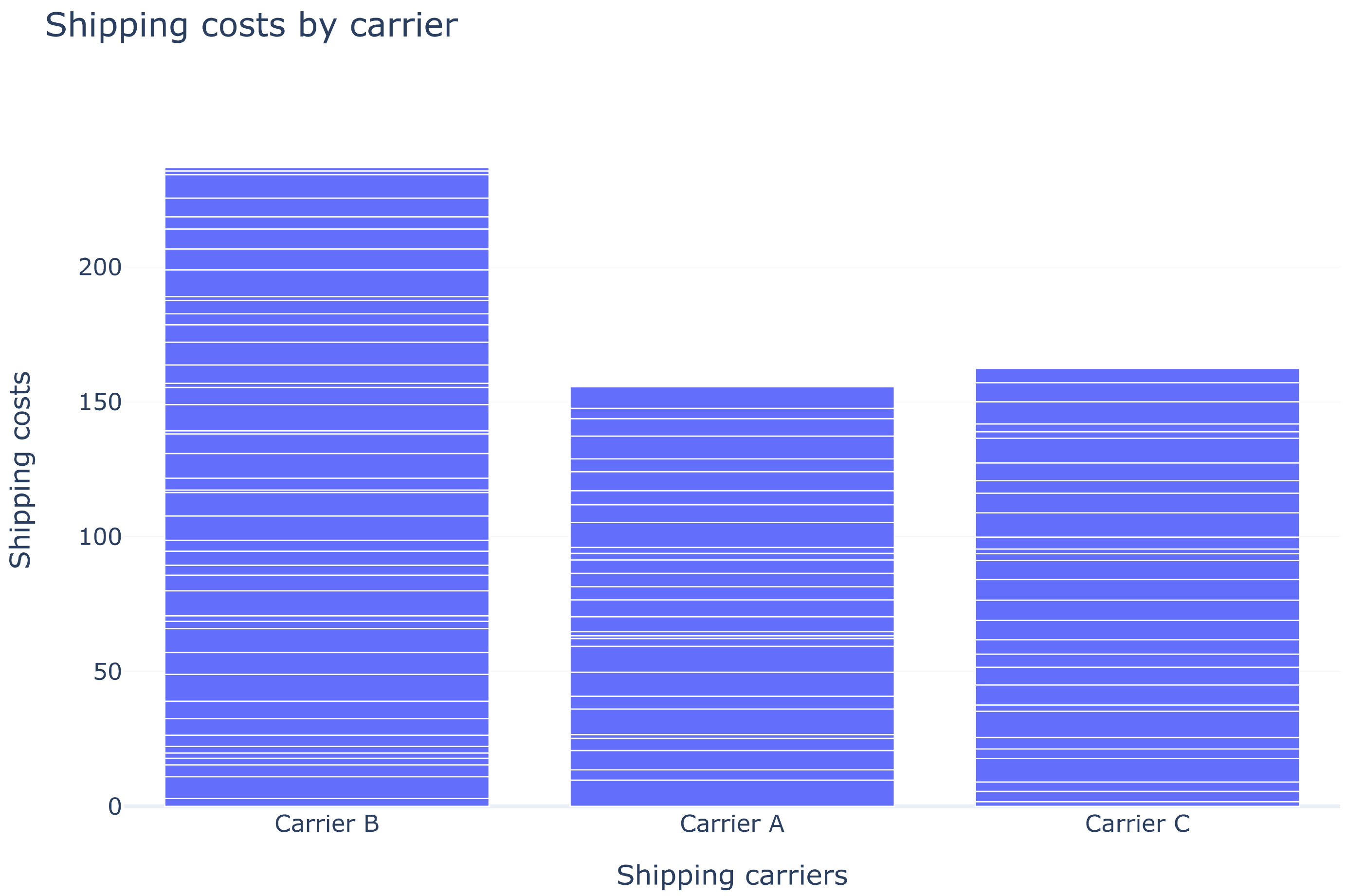
Order quantity of each SKU

```
In [19]: order_quantity_chart = px.bar(data, x='SKU', y='Order quantities',
                                       title='Order Quantity by SKU')
         order_quantity_chart.show()
```



Analyze the Shipping cost of Carriers:

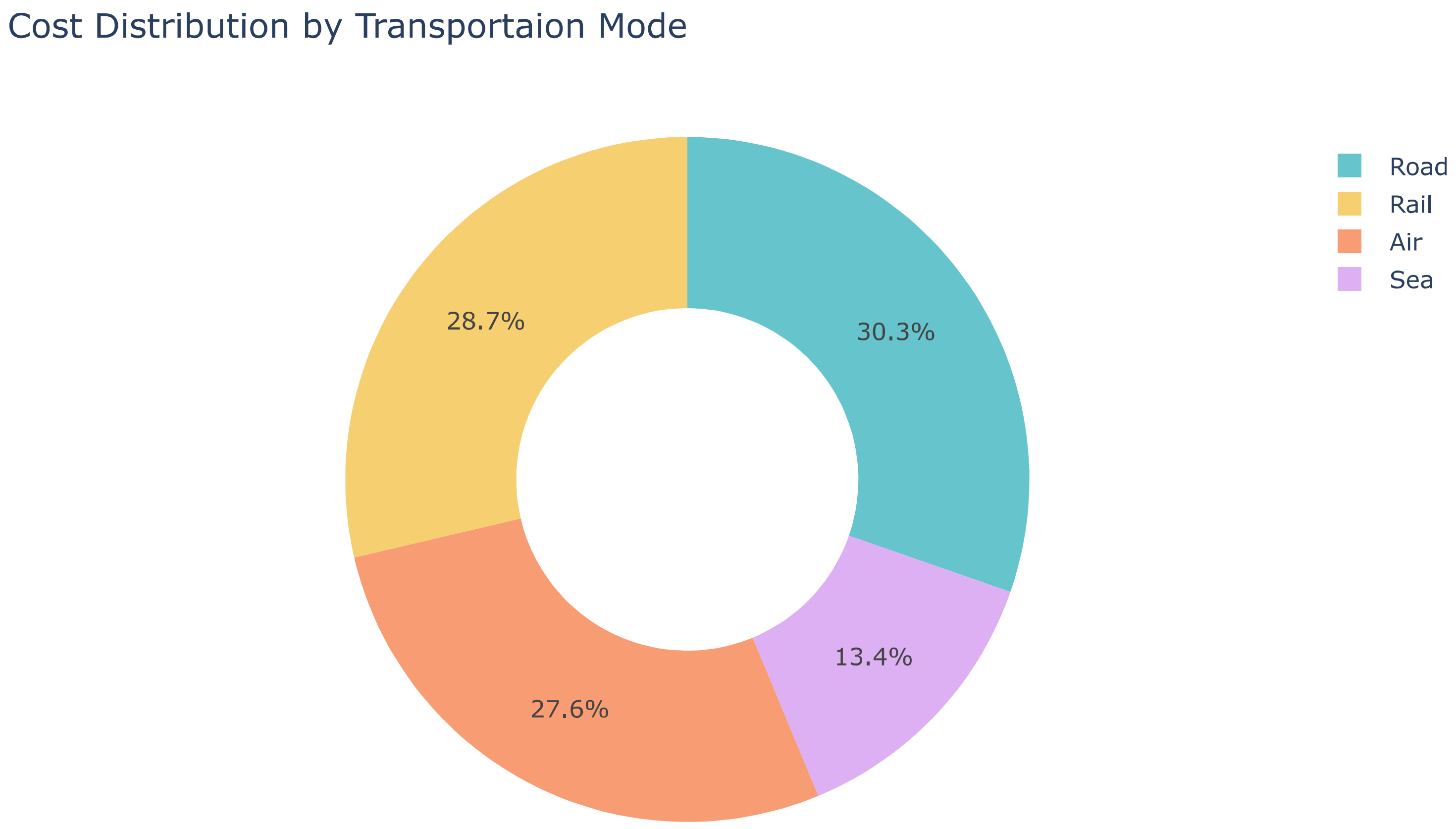
```
In [20]: shipping_cost_chart = px.bar(data, x='Shipping carriers',
                              y='Shipping costs',
                              title = 'Shipping costs by carrier')
         shipping_cost_chart.show()
```



Analyze the Transportation cost of Carriers:

In [21]:

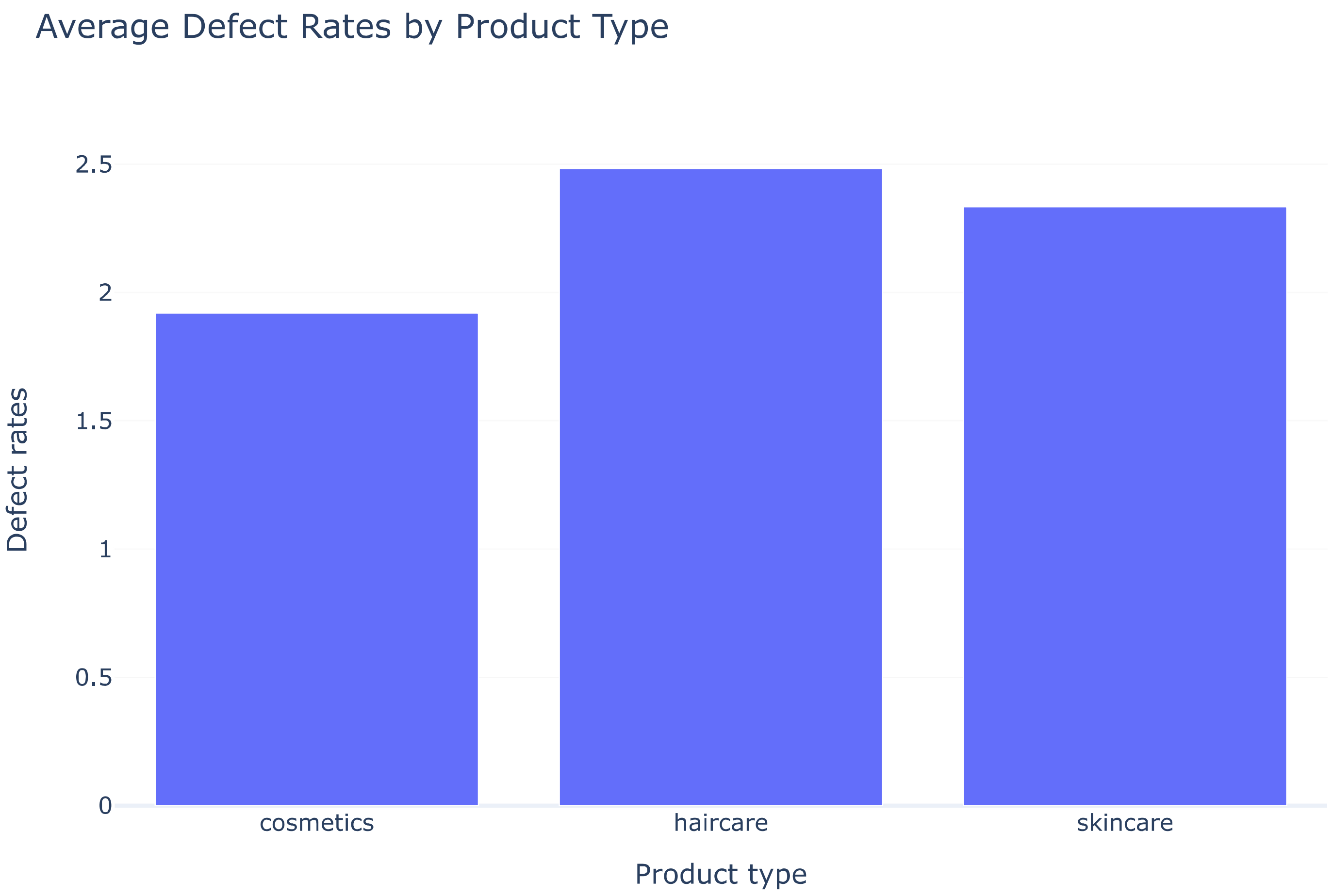
```
transportation_chart = px.pie(data,
                             values='Costs',
                             names = 'Transportation modes',
                             title='Cost Distribution by Transportaion Mode',
                             hole=0.5,
                             color_discrete_sequence=px.colors.qualitative.Pastel)
transportation_chart.show()
```



Analyse Defect Rate

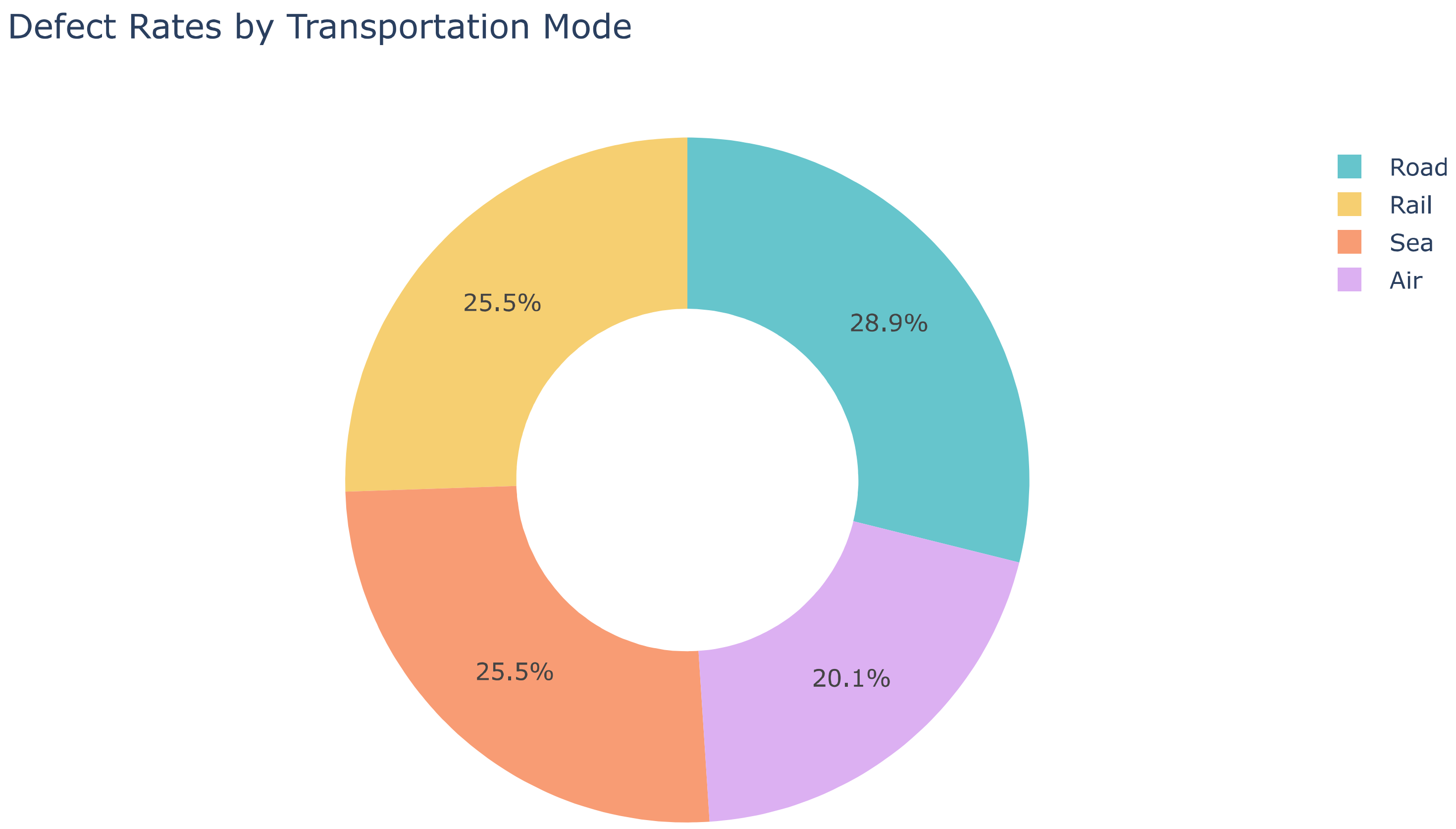
In [22]:

```
defect_rates_by_product = data.groupby('Product type')['Defect rates'].mean().reset_index()
fig = px.bar(defect_rates_by_product, x= 'Product type', y='Defect rates',
             title='Average Defect Rates by Product Type')
fig.show()
```



Defect rates by mode of transportation

```
In [24]: pivot_table = pd.pivot_table(data, values='Defect rates',
                                      index=['Transportation modes'],
                                      aggfunc='mean')
         transportation_chart = px.pie(values = pivot_table["Defect rates"],
                                       names = pivot_table.index,
                                       title='Defect Rates by Transportation Mode',
                                       hole=0.5,
                                       color_discrete_sequence=px.colors.qualitative.Pastel)
         transportation_chart.show()
```



In [ ]:

```

```