Day 29

Supply Chain Analysis Project

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")
data.head()
```

Out[2]:

	Product type	SKU	Price	Availability	Number of products sold	Revenue generated	Customer demographics	Stock levels	Lead times	q
0	haircare	SKU0	69.808006	55	802	8661.996792	Non-binary	58	7	
1	skincare	SKU1	14.843523	95	736	7460.900065	Female	53	30	
2	haircare	SKU2	11.319683	34	8	9577.749626	Unknown	1	10	
3	skincare	SKU3	61.163343	68	83	7766.836426	Non-binary	23	13	
4	skincare	SKU4	4.805496	26	871	2686.505152	Non-binary	5	3	

5 rows × 24 columns

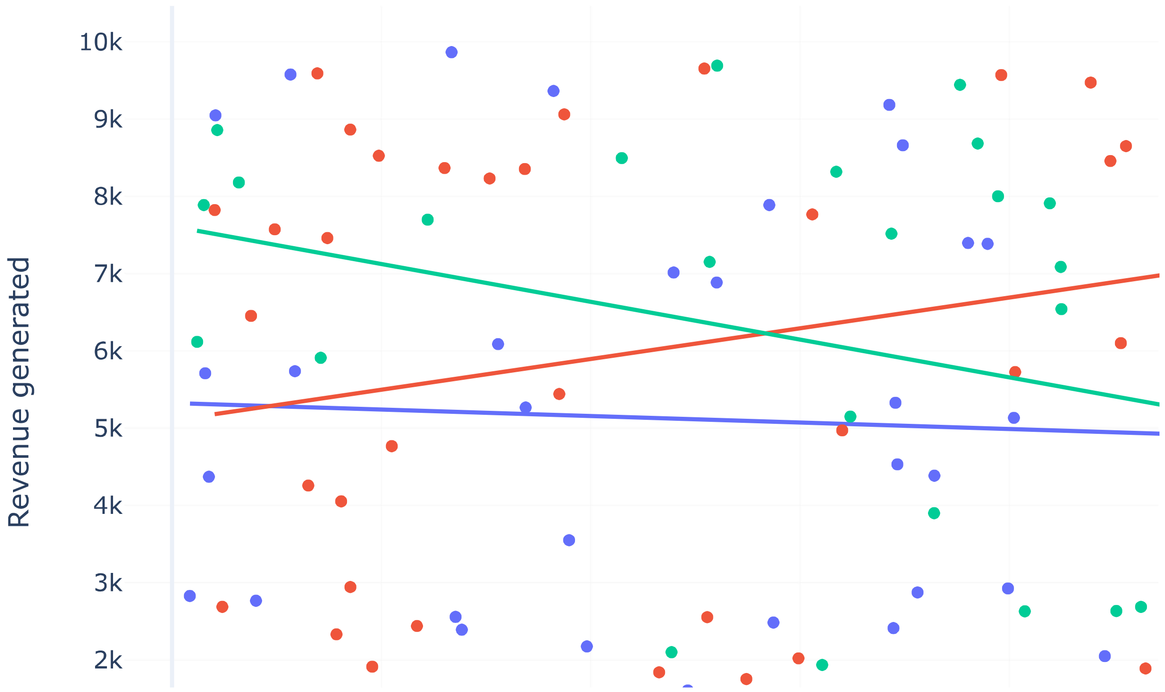
In [3]:
```
data.describe()
```

Out[3]:

	Price	Availability	Number of products sold	Revenue generated	Stock levels	Lead times	Order quantities	S
count	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	10
mean	49.462461	48.400000	460.990000	5776.048187	47.770000	15.960000	49.220000	5
std	31.168193	30.743317	303.780074	2732.841744	31.369372	8.785801	26.784429	2
min	1.699976	1.000000	8.000000	1061.618523	0.000000	1.000000	1.000000	1
25%	19.597823	22.750000	184.250000	2812.847151	16.750000	8.000000	26.000000	3
50%	51.239831	43.500000	392.500000	6006.352023	47.500000	17.000000	52.000000	6
75%	77.198228	75.000000	704.250000	8253.976921	73.000000	24.000000	71.250000	8
max	99.171329	100.000000	996.000000	9866.465458	100.000000	30.000000	96.000000	10

relationship between the price of the products and the revenue generated by them

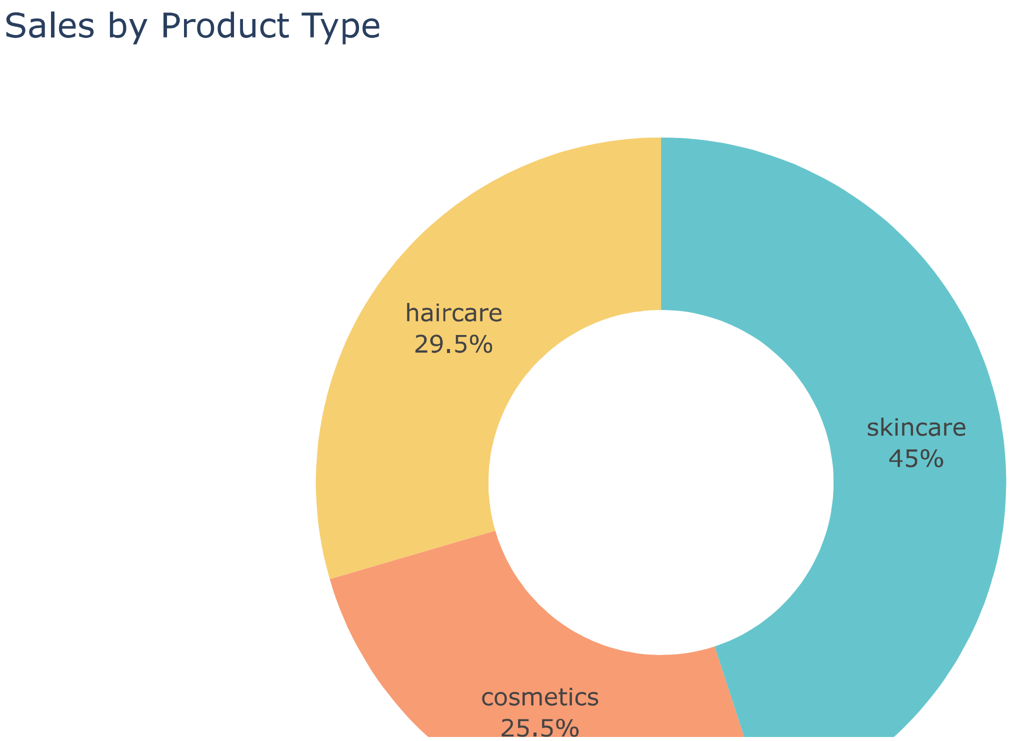
```
In [4]: fig = px.scatter(data, x='Price',
                         y='Revenue generated',
                         color='Product type',
                         hover_data=['Number of products sold'],
                         trendline="ols")
        fig.show()
```



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

Now let’s have a look at the sales by product type:

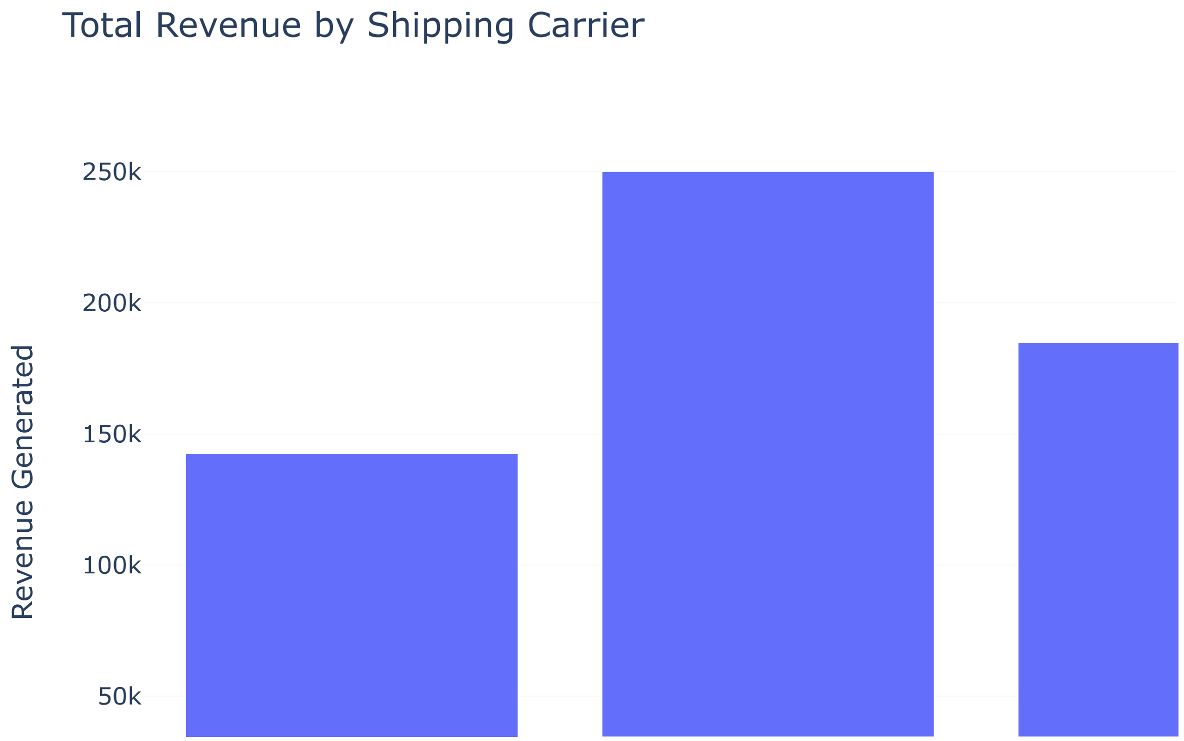
In [5]:
```
sales_data = data.groupby('Product type')['Number of products sold'].sum().res

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,
                   color_discrete_sequence=px.colors.qualitative.Pastel)

pie_chart.update_traces(textposition='inside', textinfo='percent+label')
pie_chart.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 [6]: total_revenue = data.groupby('Shipping carriers')['Revenue generated'].sum().r
        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()
```



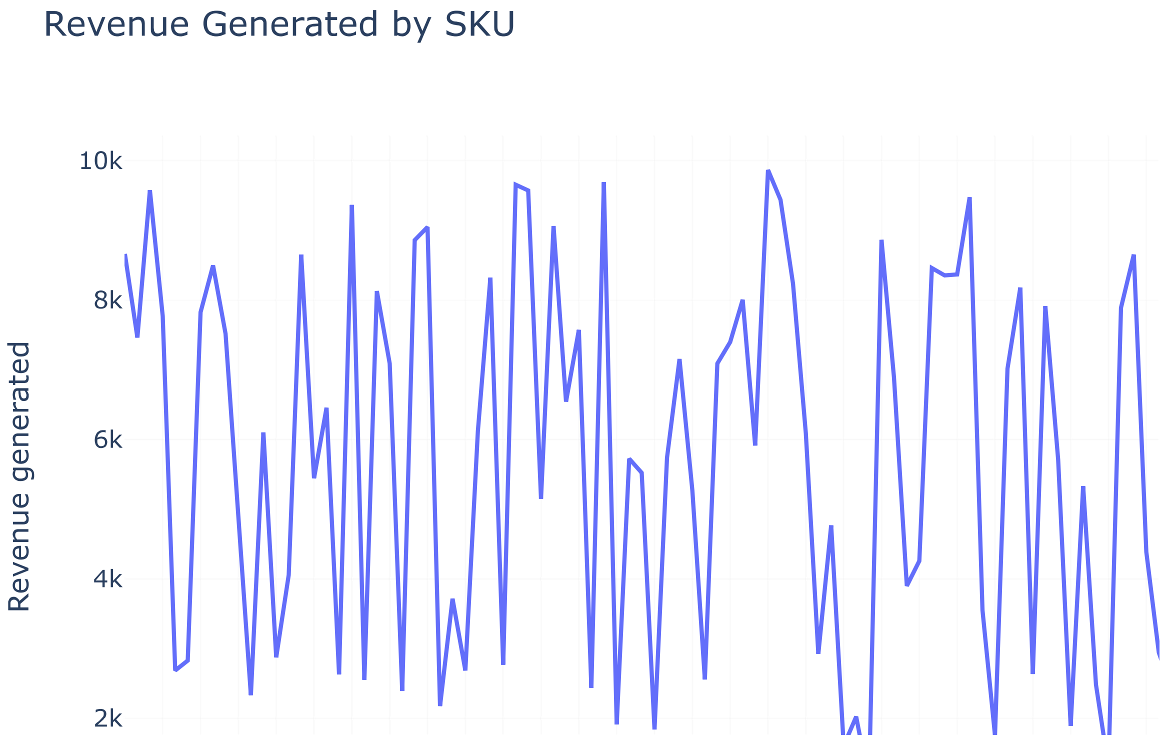
So the company is using three carriers for transportation, and Carrier B helps the company in generating more revenue

There's a column in the dataset as SKUs. You must have heard it for the very first time. So, 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 you 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.

let’s analyze the revenue generated by each SKU:

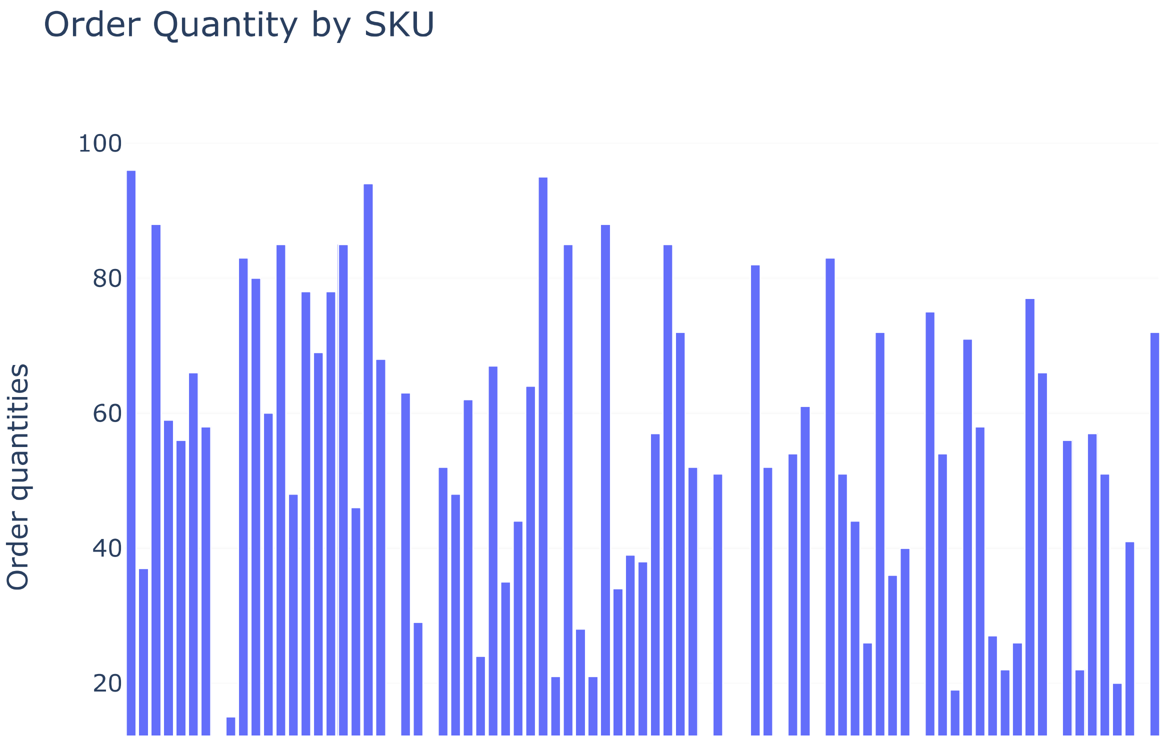
There’s a column in the dataset as SKUs. You must have heard it for the very first time. So, SKU stands for Stock Keeping Units. They’re like special codes that help companies keep track of all the different things they have for sale. Imagine you have a large toy store with lots of toys. Each toy is 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.

```
In [10]: revenue_chart = px.line(data, x='SKU',
                                 y='Revenue generated',
                                 title='Revenue Generated by SKU')
         revenue_chart.show()
```



Now let’s have a look at the order quantity of each SKU

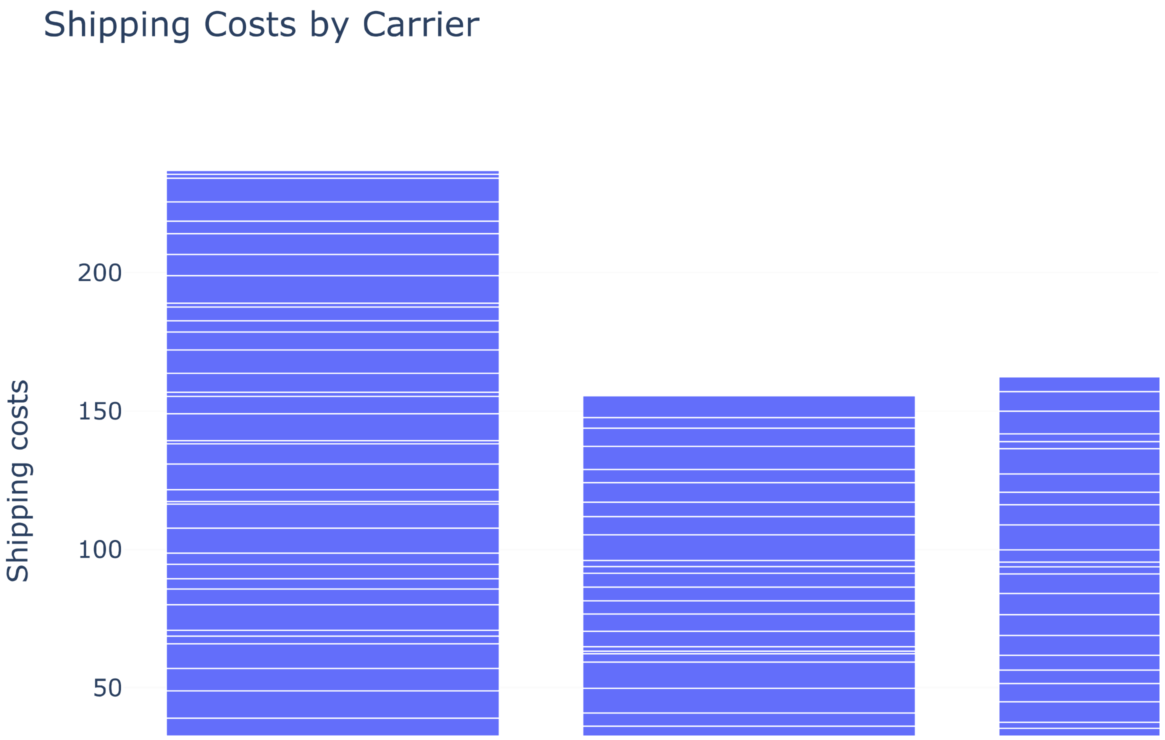
```
In [11]: order_quantity_chart = px.bar(data, x='SKU',
                                       y='Order quantities',
                                       title='Order Quantity by SKU')
         order_quantity_chart.show()
```



Now let’s analyze the shipping cost of Carriers:

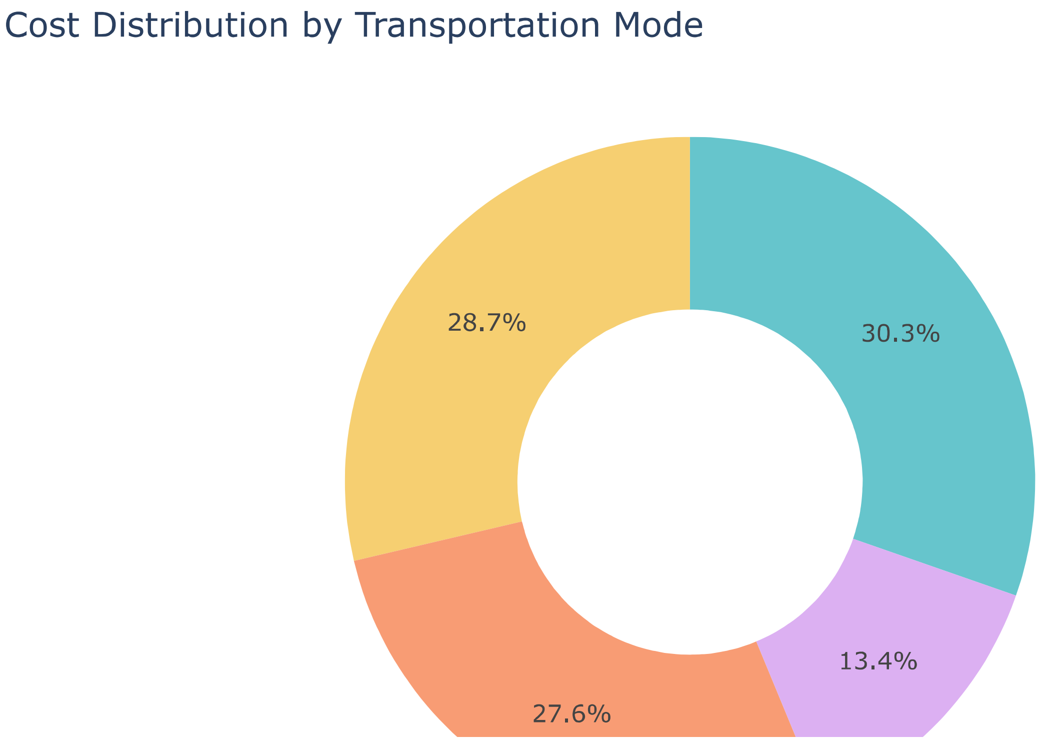
In [12]:

```
shipping_cost_chart = px.bar(data, x='Shipping carriers',
                             y='Shipping costs',
                             title='Shipping Costs by Carrier')
shipping_cost_chart.show()
```



cost distribution by transportation mode:

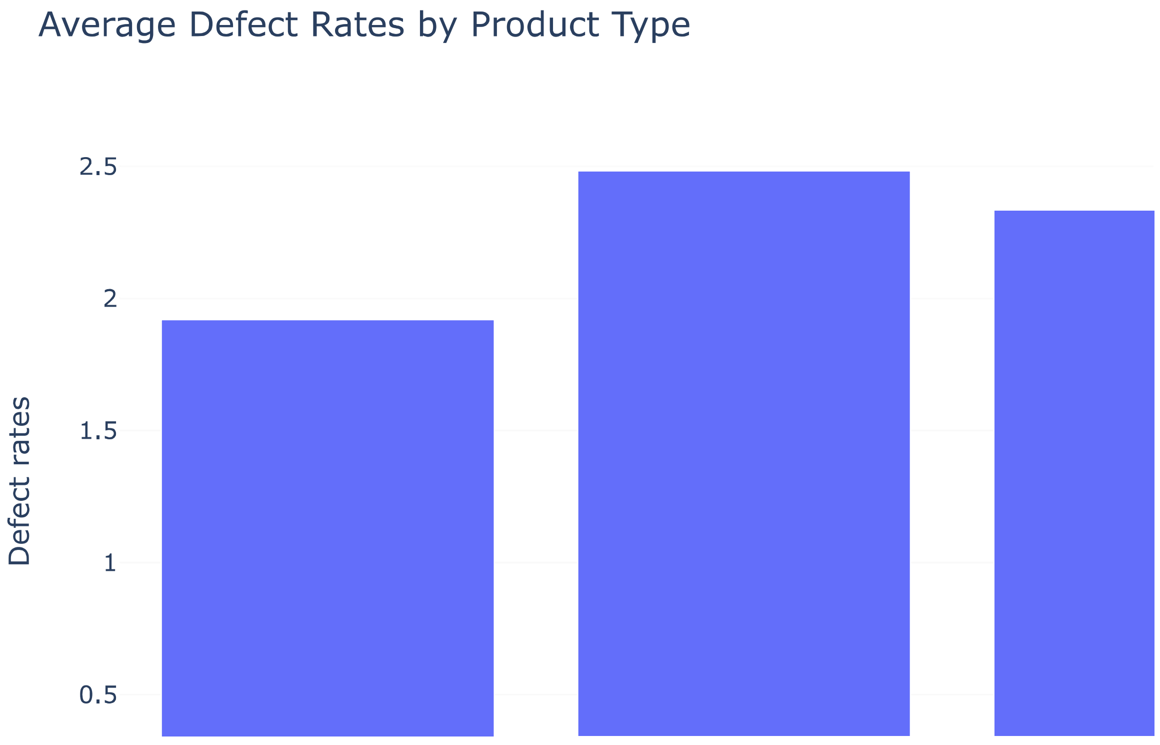
```
In [13]: transportation_chart = px.pie(data,
                              values='Costs',
                              names='Transportation modes',
                              title='Cost Distribution by Transportation Mode'
                              hole=0.5,
                              color_discrete_sequence=px.colors.qualitative.Pa
         transportation_chart.show()
```



Analyzing Defect Rate

In [14]:

```
defect_rates_by_product = data.groupby('Product type')['Defect rates'].mean().

fig = px.bar(defect_rates_by_product, x='Product type', y='Defect rates',
             title='Average Defect Rates by Product Type')
fig.show()
```



In [ ]: