```
import plotly.graph_objects as go
```

Q1. Load the "titanic" dataset using the load_dataset function of seaborn. Use Plotly express to plot a scatter plot for age and fare columns in the titanic dataset.

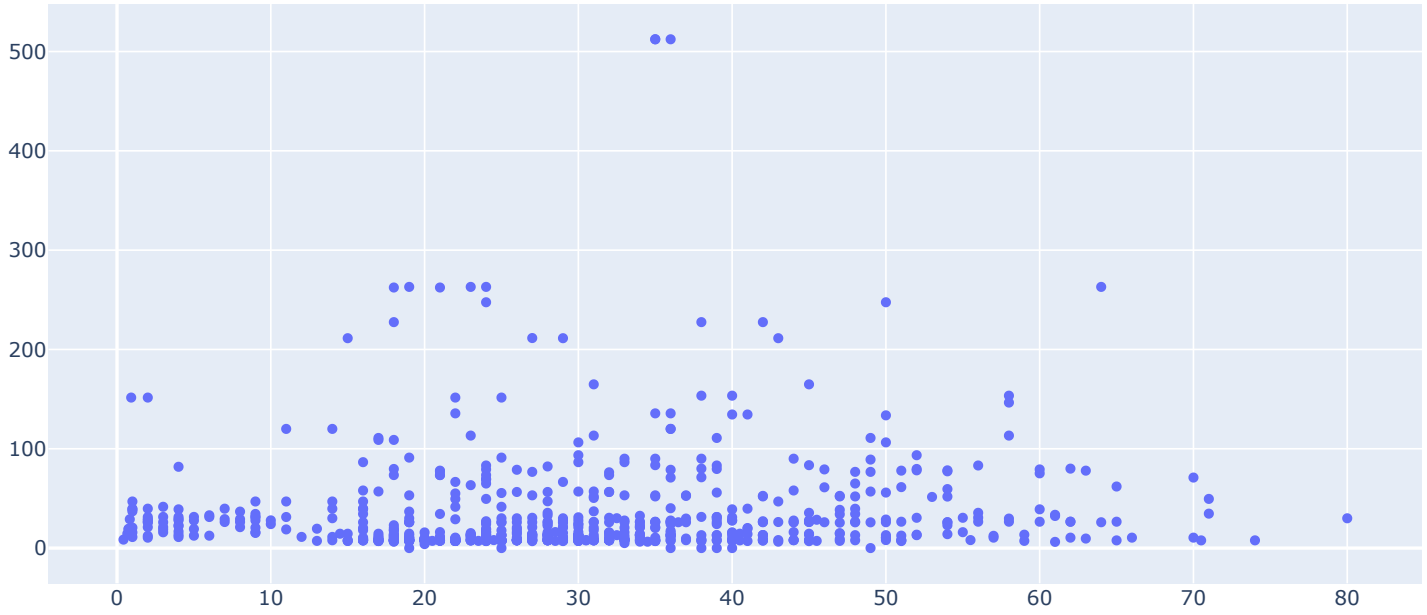
```
import seaborn as sns
titanic_data = sns.load_dataset("titanic")
```

```
titanic_data
```

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	deck	embark_town	alive	alone
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	NaN	Southampton	no	False
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	C	Cherbourg	yes	False
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	NaN	Southampton	yes	True
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	C	Southampton	yes	False
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	NaN	Southampton	no	True
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
886	0	2	male	27.0	0	0	13.0000	S	Second	man	True	NaN	Southampton	no	True
887	1	1	female	19.0	0	0	30.0000	S	First	woman	False	B	Southampton	yes	True
888	0	3	female	NaN	1	2	23.4500	S	Third	woman	False	NaN	Southampton	no	False
889	1	1	male	26.0	0	0	30.0000	C	First	man	True	C	Cherbourg	yes	True
890	0	3	male	32.0	0	0	7.7500	Q	Third	man	True	NaN	Queenstown	no	True

891 rows × 15 columns

```
fig = go.Figure()
fig.add_trace(go.Scatter(x = titanic_data.age, y = titanic_data.fare, mode = 'markers'))
fig.show()
```



Q2. Using the tips dataset in the Plotly library, plot a box plot using Plotly express.

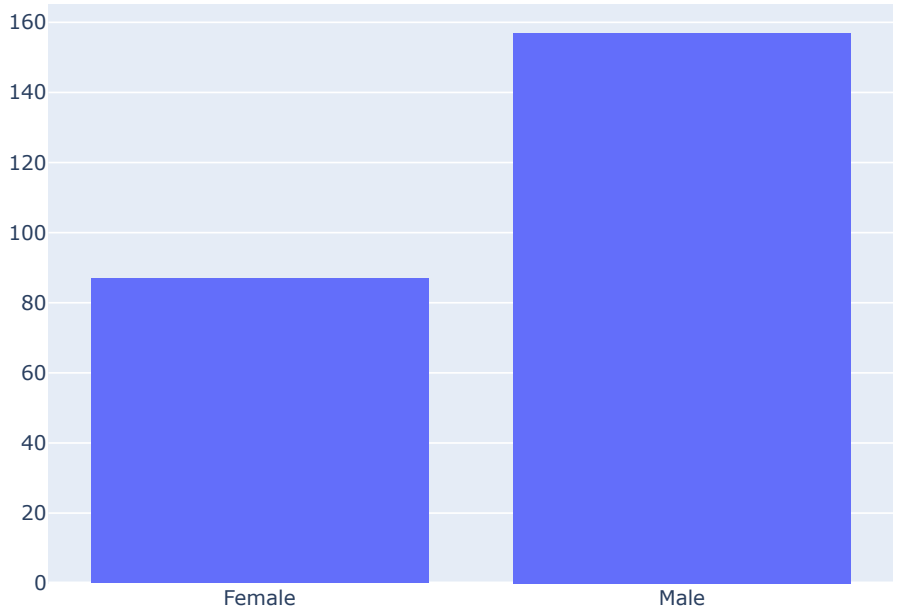
```
tips = sns.load_dataset("tips")
tips
```

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4
...	...	...	...	...	...	...	...

```
fig = go.Figure()
fig.add_trace(go.Box(y = tips.total_bill))
fig.add_trace(go.Box(y = tips.tip))
fig.show()
```

Q3. Using the tips dataset in the Plotly library, Plot a histogram for x= "sex" and y="total_bill" column in the tips dataset. Also, use the "smoker" column with the pattern_shape parameter and the "day" column with the color parameter.

```
fig = go.Figure(data = [go.Histogram(x=tips.sex, y = tips.total_bill)])
fig.show()
```



Q4. Using the iris dataset in the Plotly library, Plot a scatter matrix plot, using the "species" column for the color parameter.

```
iris = sns.load_dataset("iris")
```

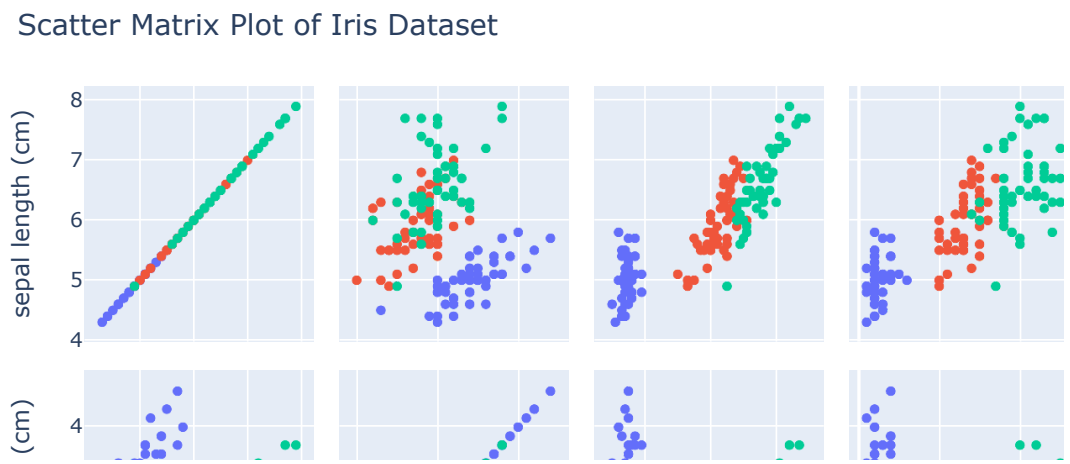
```
iris
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	virginica
146	6.3	2.5	5.0	1.9	virginica
147	6.5	3.0	5.2	2.0	virginica
148	6.2	3.4	5.4	2.3	virginica
149	5.9	3.0	5.1	1.8	virginica

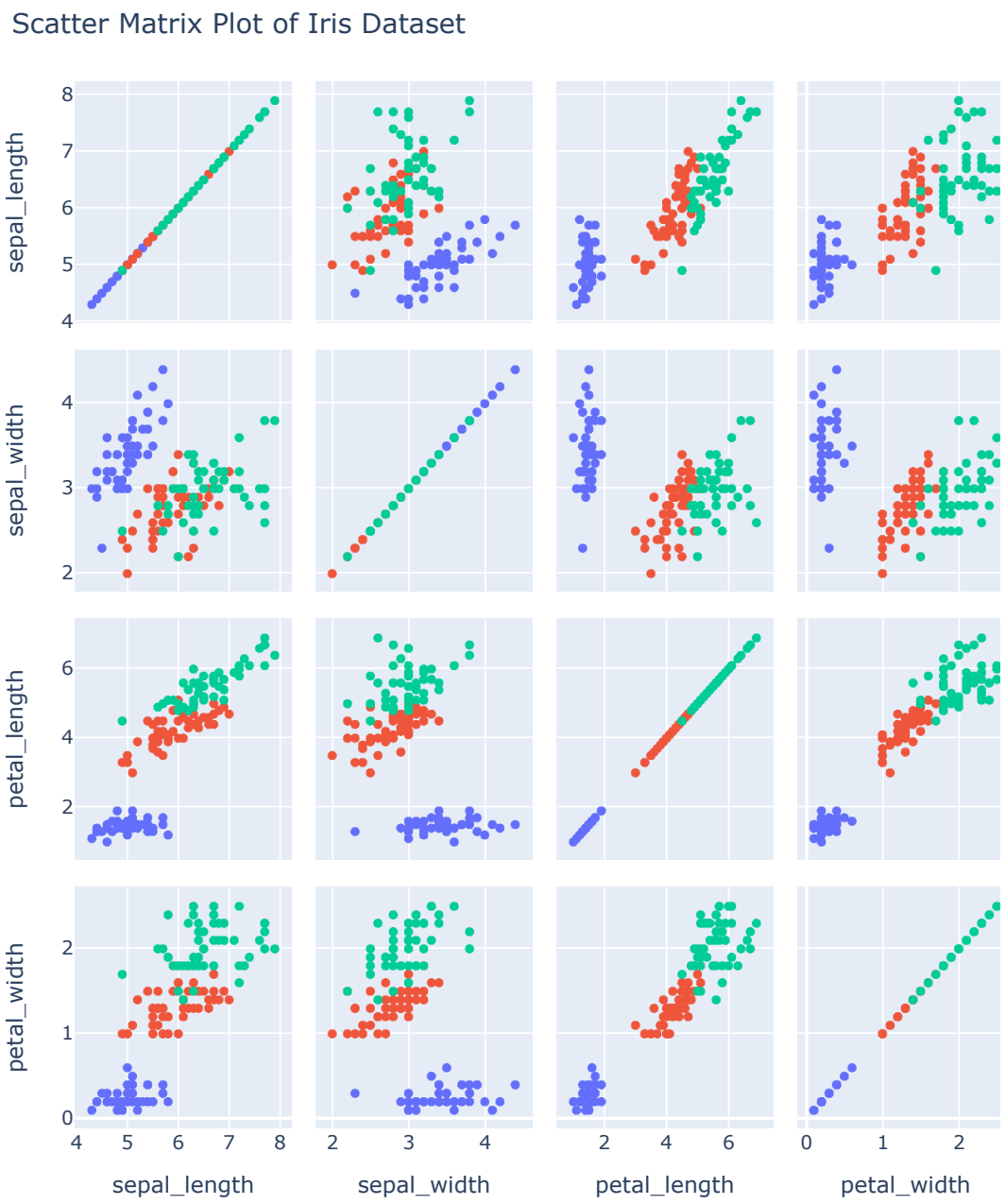
150 rows × 5 columns

```
import plotly.express as px
from sklearn.datasets import load_iris

# Load the iris dataset
iris = load_iris()
iris_data = iris.data
iris_target = iris.target
iris_species = iris.target_names[iris_target]

# Create a DataFrame from the iris data
import pandas as pd
```

```
df = pd.DataFrame(iris_data, columns=iris.feature_names)
df['species'] = iris_species

# Create the scatter matrix plot
fig = px.scatter_matrix(df, dimensions=iris.feature_names, color='species')

# Update the layout
fig.update_layout(
    title='Scatter Matrix Plot of Iris Dataset',
    width=800,
    height=800
)

# Display the plot
fig.show()
```



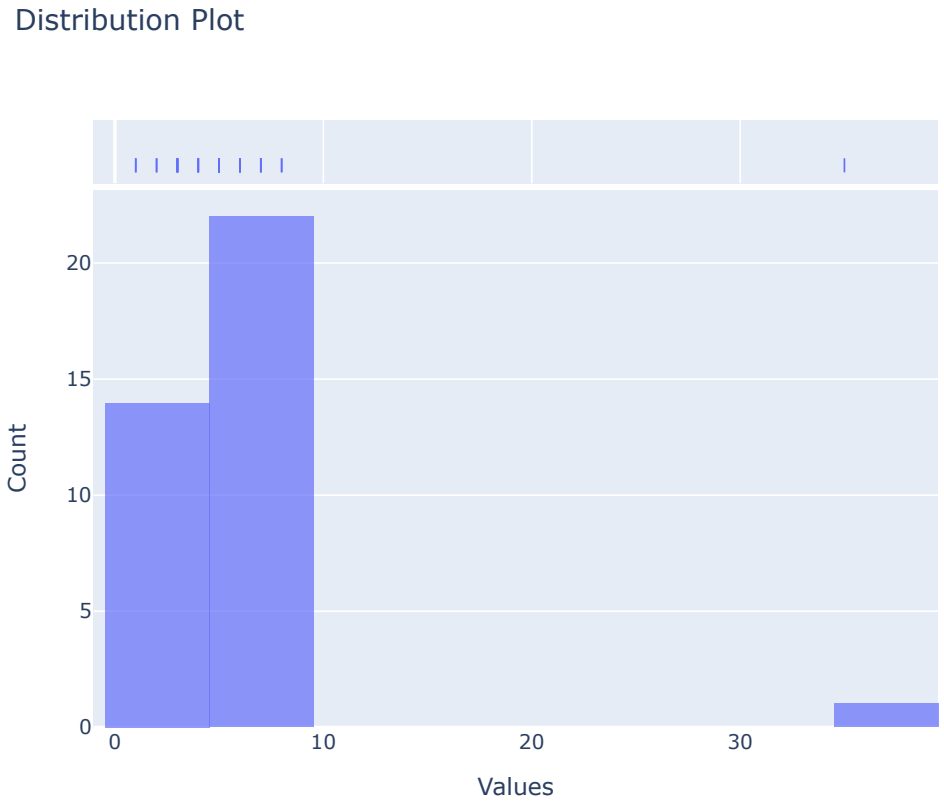

```
import plotly.express as px
from sklearn.datasets import load_iris

# Load the iris dataset
iris = load_iris()
iris_data = iris.data[:, :4]  # Selecting only the specified columns
iris_target = iris.target
iris_species = iris.target_names[iris_target]

# Create a DataFrame from the iris data
import pandas as pd
df = pd.DataFrame(iris_data, columns=["sepal_length", "sepal_width", "petal_length", "petal_width"])
df['species'] = iris_species

# Create the scatter matrix plot
fig = px.scatter_matrix(df, dimensions=["sepal_length", "sepal_width", "petal_length", "petal_width"], color='species')

# Update the layout
fig.update_layout(
    title='Scatter Matrix Plot of Iris Dataset',
    width=800,
    height=800
)

# Display the plot
fig.show()
```



Q5. What is Distplot? Using Plotly express, plot a distplot.

In Plotly Express, the distplot function is used to create a histogram with a smooth density curve overlaid on it. It allows you to visualize the distribution of a single variable.

```
import plotly.express as px
import pandas as pd

data = pd.DataFrame({'values': [1,2,3,1,3,35,2,6,5,3,6,5,5,8,8,6,3,5,7,5,3,6,7,5,5,7,4,3,6,5,4,5,5,4,5,4,5]})
fig = px.histogram(data, x="values", marginal="rug", opacity=0.7)
fig.update_layout(
    title='Distribution Plot',
    xaxis_title='Values',
    yaxis_title='Count',
    showlegend=False
)
fig.show()
```



s here

✓ 0s completed at 3:40 PM