Exploratory Data Analysis - Part One

Descriptive Analysis in Python

- describe()

```
df_operations.describe()
```

	Customer	Purchases	Sales	Refunds
count	19.000000	19.000000	19.000000	19.000000
mean	10009.000000	450589.210526	563252.210526	819.210526
std	5.627314	167280.787361	209101.355900	439.467554
min	10000.000000	83000.000000	103750.000000	0.000000
25%	10004.500000	388850.000000	486062.500000	592.000000
50%	10009.000000	454100.000000	567925.000000	910.000000
75%	10013.500000	531200.000000	664000.000000	1062.500000
max	10018.000000	741000.000000	926250.000000	1482.000000

Descriptive Analysis in Python

- describe()

```
df_operations.describe(include = [np.object])
```

	Customer Type	Payment Type	Country	Continent
count	19	19	19	19
unique	2	3	8	3
top	Company	Cash	EEUU	America
freq	10	8	6	14

Descriptive Analysis in Python

- value_counts()

```
df_cus_type_counts = df_operations["Customer Type"].value_counts()
df_cus_type_counts.rename(columns={'Customer Type':'Count'}, inplace = True)
df_cus_type_counts.index.name = 'Customer Type'
```

Customer Type	Count
Company	10
Person	9

Descriptive Analysis in Python

- groupby()

```
df_test = df_operations[['Customer Type','Payment Type','Sales']]
df_test.groupby(['Customer Type','Payment Type'], as_index = False).mean()
```

	Customer Type	Payment Type	Sales
0	Company	Cash	530185.000000
1	Company	Credit Card	711425.666667
2	Company	Transfer	503756.000000
3	Person	Cash	547179.000000
4	Person	Credit Card	508875.000000
5	Person	Transfer	562447.750000

Descriptive Analysis in Python

- pivot()

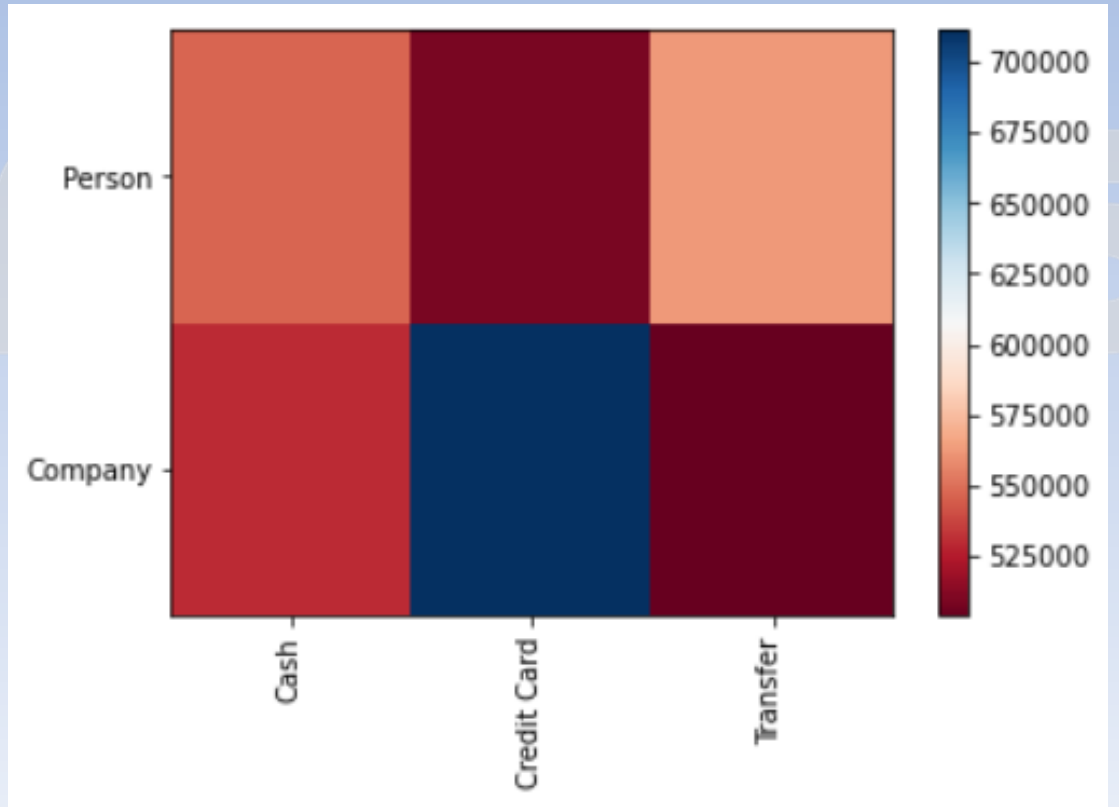
```
df_grp = df_test.groupby(['Customer Type','Payment Type'], as_index = False).mean()
df_grp.pivot(index = 'Customer Type', columns = 'Payment Type')
```

	Sales		
Payment Type	**Cash**	**Credit Card**	**Transfer**
Customer Type			
Company	530185.0	711425.666667	503756.00
Person	547179.0	508875.000000	562447.75

Descriptive Analysis in Python

❑ Heatmap

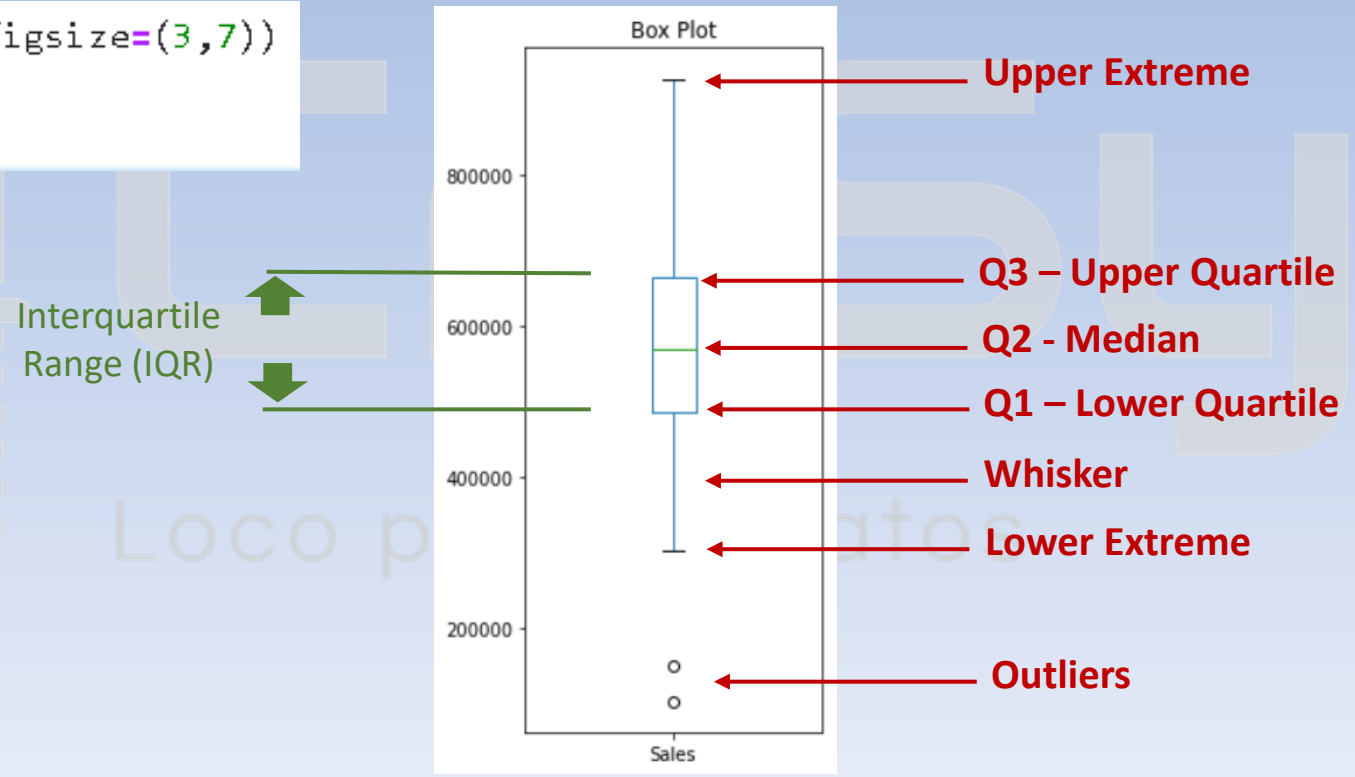
```
df_pivot = df_grp.pivot(index = 'Customer Type', columns = 'Payment Type')
plt.pcolor(df_pivot, cmap = 'RdBu')
plt.colorbar()
plt.show()
```



Descriptive Analysis in Python

❑ Box Plots with matplotlib library

```
df_oper_sales.plot(kind='box', figsize=(3,7))
plt.title('Box Plot')
plt.show()
```



Descriptive Analysis in Python

- Box Plots with seaborn library.

```
sns.boxplot(x="Continent", y="Sales", data=df_operations)
```