

### Mini project on EDA and Visualisation

```
df=pd.read_csv('Data.csv')
```

```
df.head()
```

```
s=df.loc[:3,:]      #including the outer point as well, means will print row 0,1,2,3
```

```
A1 = s.to_dict('list')
```

```
A2=str(df['Currency_Name'][df['Price']==df['Price'].max()][0])
```

```
A3=df['Change(24h)'].mean()
```

```
df1=df[(df['Market_Capital']>20000000000) & (df['Price']>=200)]
```

```
a=df1.sort_values(by='Currency_Name', ascending=True)
```

```
A4=list(a['Currency_Name'])
```

```
A5=float(df['Volume(24h)'].min())
```

**#Write your code here**

```
fig, ax = plt.subplots(figsize=(15,8))
```

**# Plot the Heatmap with axes (ax) and assign the returned object(path collection) to q6**

```
s=df[['Rank','Market_Capital','Price','Circulating_Supply','Volume(24h)','Change(24h)']]
```

```
co=s.corr(method='pearson')
```

```
q6 = sns.heatmap(co, annot=True)
```

```
ax.set_xlabel('X-axis')
```

```
ax.set_ylabel('Y-axis')
```

```
ax.set_title('Heatmap Correlation')
```

**#Assigning the object to A6 for validation**

```
A6=q6
```

**#Write your code here**

```
fig7, ax7 = plt.subplots(figsize=(8,8))
```

**# Plot the countplot with axes (ax7) and assign the returned object(path collection) to q7**

```
#df1= df.groupby('Currency_Name').count()
```

```
s=df.sort_values(by='Market_Capital', ascending=False)
```

```
s=s[:10]
```

```
q7 = sns.barplot(data=s, x='Market_Capital', y='Currency_Name')
```

```
ax7.set_title("Barplot containing Top10 Currency names with highest market capital")
```

**#Assigning the object,axes to A7 for validation**

```
A7=[q7,ax7]
```

**#Write your code here**

```
fig8, ax8 = plt.subplots(figsize=(8,8))
```

**# Plot the violinplot with axes (ax8) and assign the returned object(path collection) to q8**

```
q8 = sns.violinplot(data=df,y='Change(24h)')
```

```
ax8.set(xlabel='X_axis', ylabel='Y_axis', title='violinplot for Change(24h)')
```

**#Assigning the object to A8 for validation**

```
A8=q8
```

**#Write your code here**

```
fig9, ax9 = plt.subplots(figsize=(8,8))
```

**# Plot the scatterplot with axes (ax9) and assign the returned object(path collection) to q9**

```
s=df.sort_values(by='Market_Capital', ascending=False)
```

```
s=s[:50]
```

```
q9 = sns.scatterplot(data=s, x='Market_Capital', y='Price')
```

```
ax9.set_title('Scatterplot between Price and Market Capital')
```

**#Assigning the object to A9 for validation**

```
A9=q9
```

**#Write your code here**

```
fig10, ax10 = plt.subplots(figsize=(8,8))
```

**# Plot the barplot with axes (ax10) and assign the returned object(path collection) to q10**

```
s=df.sort_values(by='Change(24h)', ascending=False)
```

```
s=s[:10]
```

```
q10 = sns.barplot(data=s, x='Change(24h)', y='Currency_Name')
```

```
ax10.set_title('Barplot containing Top10 Currency names with Change(24h)')
```

**#Assigning the object,axes to A10 for validation**

```
A10=[q10,ax10]
```