

# Data

## Visualization

### Assignment

Name: K. Rakesh

VIU No: 22710

Slot No: S2L5



Submission To:

Dr. Kausalya. K



①  
1. Show dataset Shape, Column details, and missing values.

Code:

```
import Pandas as pd
df = Pd.read_csv("University-admission.csv")
Print("Dataset Shape:", df.shape)
Print("\n Column Details:")
Print(df.info())
Print("\n missing values:")
Print(df.isnull().sum())
```

Output:

Data set Shape : (20,6)

Column Details : <class 'Pandas.Core.Frame'>

S.No	Column	Non-Null Set	D type
0	GRE-Score	20 non-null	int64
1	GPA	20 non-null	float64
2	Admission	20 non-null	float64
3	Statement	20 non-null	object
4	Home town	20 non-null	object
5	Year	20 non-null	float64



Missing values	
GRE - Scores	0
GPA	0
Admission	0
Statement	0
Hometown	0
Year	0

dtype : int64

2. Construct histograms for GRE Scores and violin plots for GPA vs Admission Chance.

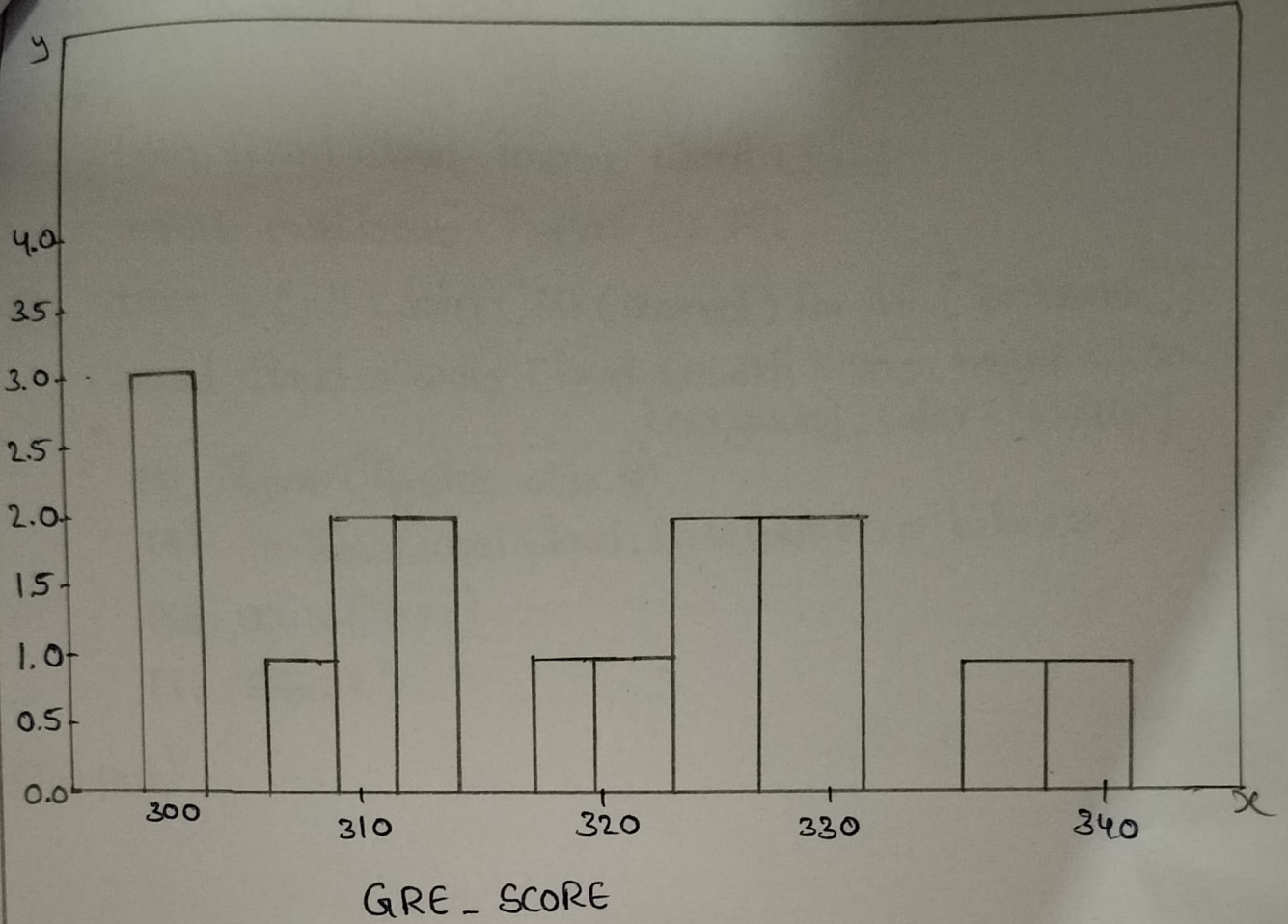
Code:

```

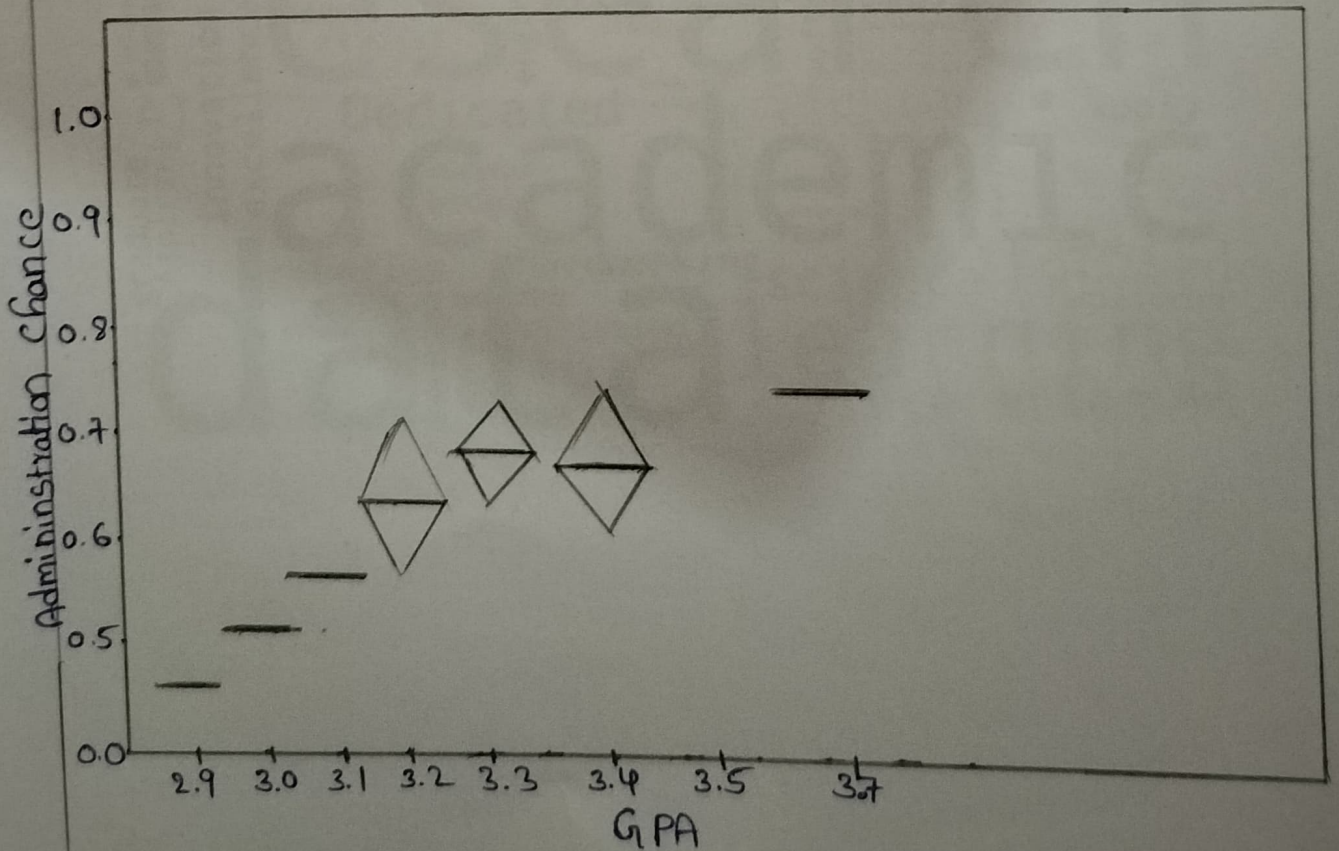
import matplotlib.pyplot as plt
import seaborn as sns
plt.figure(figsize=(7,5))
sns.histplot(df['GRE - score'], bins=20)
plt.title('Distribution of GRE Scores')
plt.show()
plt.figure(figsize=(7,5))
sns.violinplot(x='GPA', y='Admission - chance')
plt.title('GPA vs Admission Chance')
plt.show()

```





GPA VS Admission - Chance:-



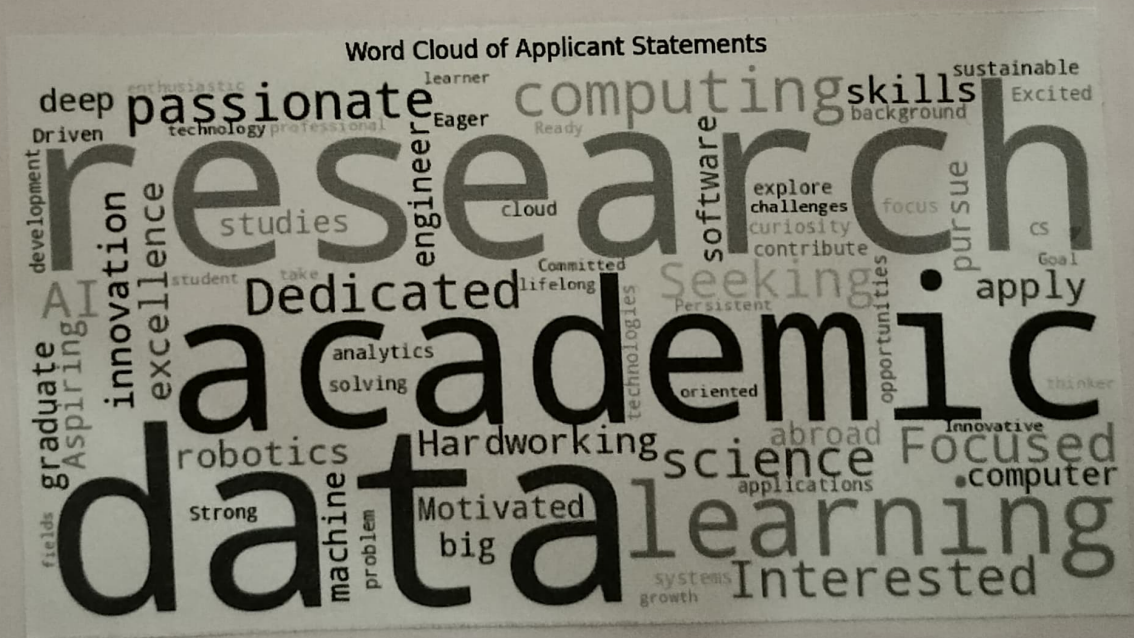


Generate a word cloud of applicant statements:

Code:-

```
from wordcloud import WordCloud
import matplotlib.pyplot as plt
text = " ".join(str(statement) for df['statement'])
wordcloud = WordCloud(width=800, height=400,
                       background_color="white")
plt.figure(figsize=(10,5))
plt.imshow(wordcloud.interpolation='bilinear')
plt.axis('off')
plt.show()
```

Output:-

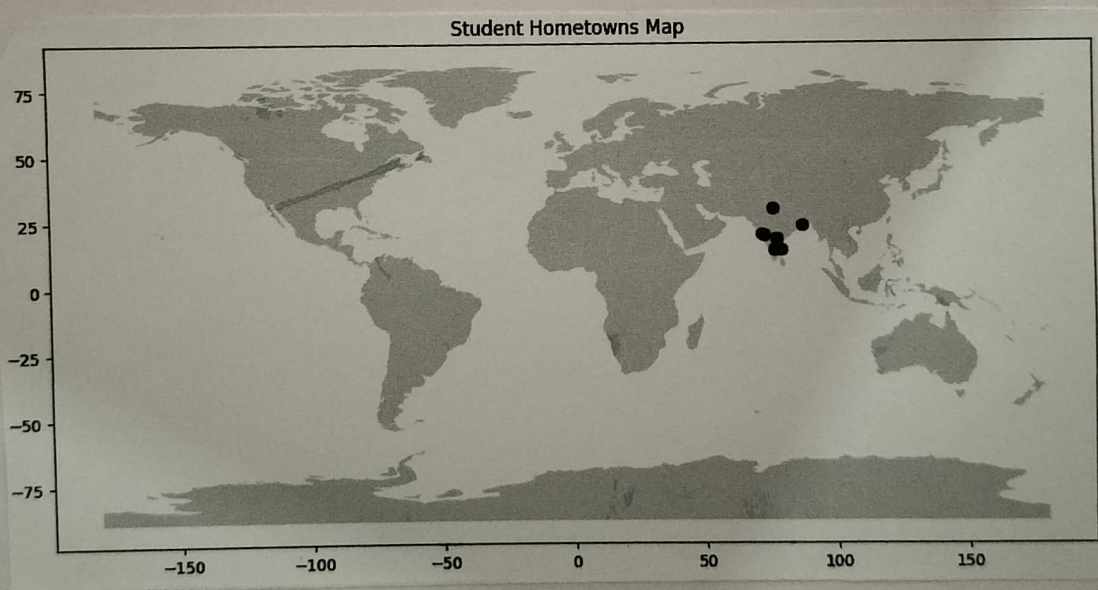




# Map Student Hometowns using geo-spatial data

Output:-

```
import geopandas as gpd
from geopandas.tools import geocode
locations = geocode(df['Home town'], Provider =
                  'nomination')
geo_df = df.join(locations)
world = gpd.read_file(gpd.datasets.get_path('natural-
lower's'))
ax = world.plot(colour='light gray', figsize=(120))
geo_df.plot(ax=ax, colour='red', marker_size=80)
plt.size title("Student Hometown Map")
plt.show()
```





Analyze yearly admission trends with line graphs:

Code:-

```
# Assuming df as column: 'year', 'Admission'
yearly_trend = df.groupby('year')['Admission']

plt.figure(figsize=(3,5))
plt.plot('yearly_trend['year]', yearly_trend
        ['Admission - Chance'])

plt.xlabel("year")
plt.ylabel("Average Admission Chance")
plt.grid(True)
plt.show()
```

Output:-

