TASK - 1

Problem Statement : Create a barchart or histogram to visualize distribution of a categorical or continuous variables

Dataset Used : Cardiovascular Disease Dataset (Kaggle)

About Dataset : This is a cardiovascular disease dataset which contains 3 types of features like , factual information(Objective),results of medical examination(Examination),information given by the patients(Subjective).This dataset include categorical as well as numerical values including binary values.

Exploratory Data Analysis(EDA)

```
In [1]: import pandas as pd
        import matplotlib.pyplot as plt
```

```
In [2]: #read the csv file into a pandas framework
        data=pd.read_csv("D:\MSc Data Science\Semester 3\Extra Works\Prodigy InfoTech\
```

In [3]:
```
#Descriptive Statistics
print(data.describe())
```

```
                id           age       gender       height        ap_hi  \
count  1000.000000   1000.000000  1000.000000  1000.000000  1000.000000   
mean    717.934000  19414.046000     1.362000   164.173000   127.414000   
std     416.244071   2532.924365     0.480819     8.326608    16.262628   
min       0.000000  14321.000000     1.000000    76.000000    90.000000   
25%     349.250000  17500.500000     1.000000   159.000000   120.000000   
50%     737.500000  19659.000000     1.000000   164.000000   120.000000   
75%    1071.500000  21363.250000     2.000000   170.000000   140.000000   
max    1429.000000  23661.000000     2.000000   188.000000   180.000000   

             ap_lo  cholesterol         gluc        smoke         alco  \
count  1000.000000  1000.000000  1000.000000  1000.000000  1000.000000   
mean     81.562000     1.390000     1.242000     0.097000     0.047000   
std       9.175421     0.698848     0.589732     0.296106     0.211745   
min      60.000000     1.000000     1.000000     0.000000     0.000000   
25%      80.000000     1.000000     1.000000     0.000000     0.000000   
50%      80.000000     1.000000     1.000000     0.000000     0.000000   
75%      90.000000     2.000000     1.000000     0.000000     0.000000   
max     120.000000     3.000000     3.000000     1.000000     1.000000   

            active       cardio    age_years          bmi  Unnamed: 16  \
count  1000.000000  1000.000000  1000.000000  1000.000000          0.0   
mean      0.778000     0.501000    52.677000    27.756585          NaN   
std       0.415799     0.500249     6.955361     5.775825          NaN   
min       0.000000     0.000000    39.000000    16.652494          NaN   
25%       1.000000     0.000000    47.000000    24.031910          NaN   
50%       1.000000     1.000000    53.000000    26.794550          NaN   
75%       1.000000     1.000000    58.000000    30.411182          NaN   
max       1.000000     1.000000    64.000000    95.221607          NaN   

       weight_before  weight_after  
count    1000.000000   1000.000000  
mean       74.686300     72.355300  
std        15.241528     15.172589  
min        42.000000     42.000000  
25%        65.000000     62.000000  
50%        72.000000     70.000000  
75%        84.000000     80.000000  
max       200.000000    200.000000  
```

In [4]:

```
data.head()
```

Out[4]:

	id	age	gender	height	ap_hi	ap_lo	cholesterol	gluc	smoke	alco	active	cardio	age_
0	0	18393	2	168	110	80	1	1	0	0	1	0	
1	1	20228	1	156	140	90	3	1	0	0	1	1	
2	2	18857	1	165	130	70	3	1	0	0	0	1	
3	3	17623	2	169	150	100	1	1	0	0	1	1	
4	4	17474	1	156	100	60	1	1	0	0	0	0	

In [5]:

```
data.tail()
```

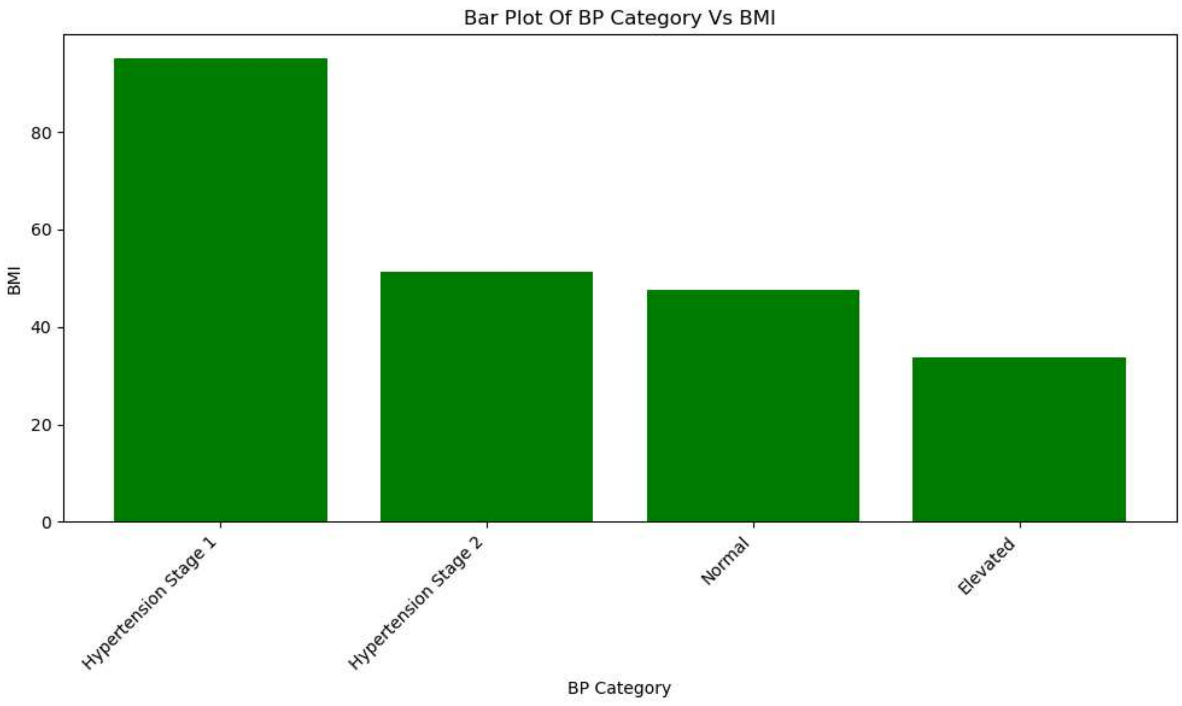
Out[5]:

	id	age	gender	height	ap_hi	ap_lo	cholesterol	gluc	smoke	alco	active	cardio
995	1421	14715	1	166	110	70	1	1	0	0	1	0
996	1423	22401	1	158	130	90	1	2	0	0	1	1
997	1426	18398	2	165	150	90	1	1	0	0	1	0
998	1427	23362	2	171	120	80	1	1	0	0	1	0
999	1429	21118	1	158	130	80	1	1	0	0	0	0

VISUALIZATION

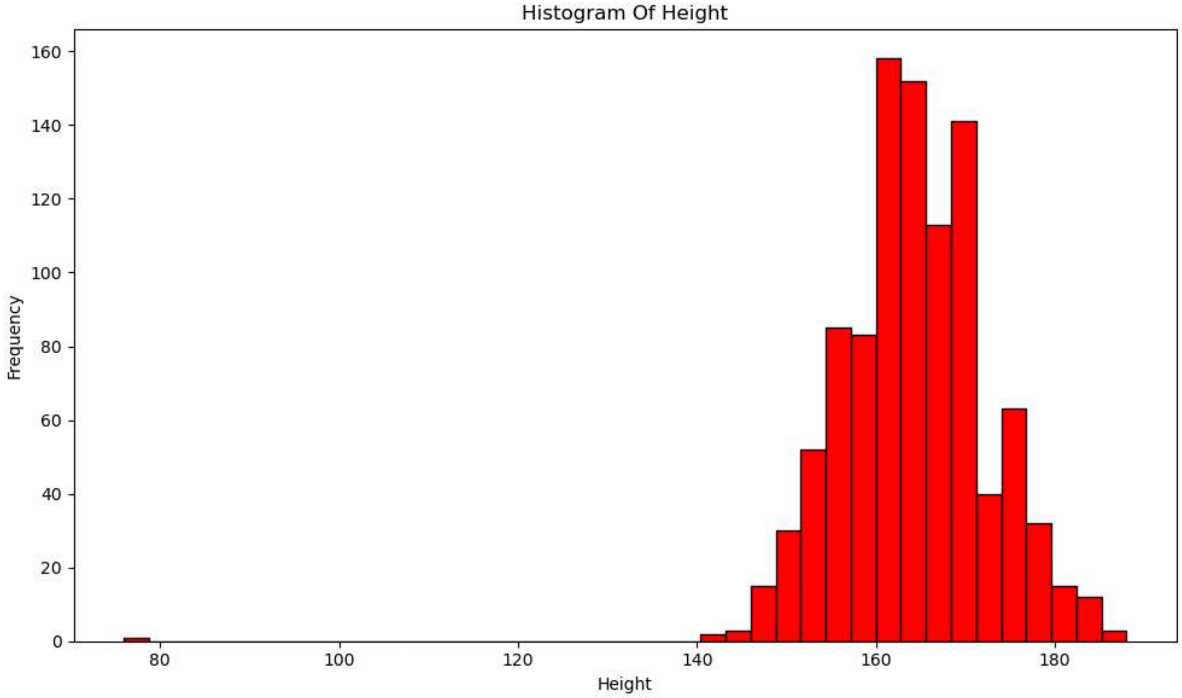
In [6]:

```
#bar plot for bp_category vs bmi
plt.figure(figsize=(10, 6))
plt.bar(data['bp_category'],data['bmi'],color='green')
plt.xlabel('BP Category')
plt.ylabel('BMI')
plt.title('Bar Plot Of BP Category Vs BMI')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()  # Adjust layout to prevent clipping of labels
plt.show()
```



In [8]:

```
#histogram for height and weight
plt.figure(figsize=(10, 6))
plt.hist(data['height'],bins=40,color='red',edgecolor='black')
plt.xlabel('Height')
plt.ylabel('Frequency')
plt.title('Histogram Of Height')
plt.tight_layout()  # Adjust layout to prevent clipping of labels
plt.show()
```


Histogram Of Height
Frequency
Height