In [1]:
```
# Import packages
```

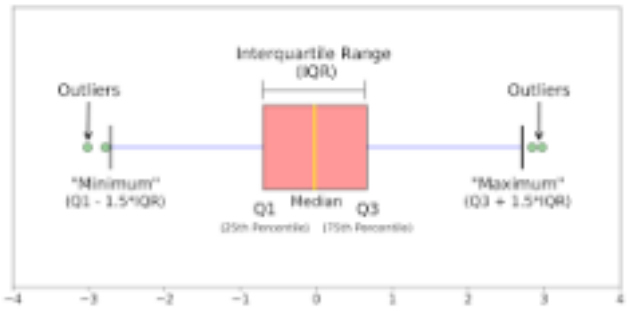
In [2]:
```
# Read the data

import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt


file_path="C:\\Users\\omkar\\OneDrive\\Documents\\Data science\\Naresh IT\\
visa_df=pd.read_csv(file_path)
visa_df
```

Out[2]:

	case_id	continent	education_of_employee	has_job_experience	requires_job_traini
0	EZYV01	Asia	High School	N	
1	EZYV02	Asia	Master's	Y	
2	EZYV03	Asia	Bachelor's	N	
3	EZYV04	Asia	Bachelor's	N	
4	EZYV05	Africa	Master's	Y	
...	...	...	...	...	
25475	EZYV25476	Asia	Bachelor's	Y	
25476	EZYV25477	Asia	High School	Y	
25477	EZYV25478	Asia	Master's	Y	
25478	EZYV25479	Asia	Master's	Y	
25479	EZYV25480	Asia	Bachelor's	Y	

25480 rows × 12 columns

??????-?????????

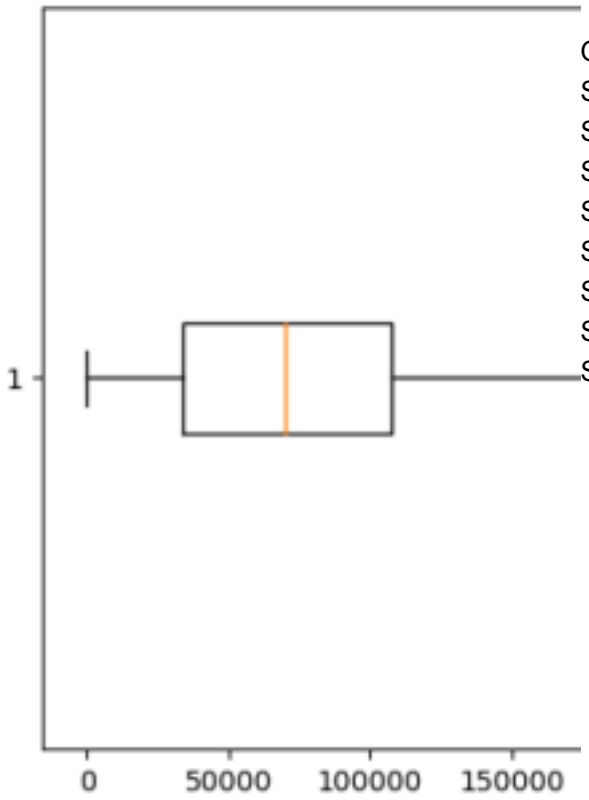


In [8]:
```
plt.boxplot(visa_df['prevailing_wage']
,
```


12/15/23, 12:30 PM EDA-5-Outlier analysis - Jupyter Notebook

```
 vert=False)
plt.show()
```

```
#black dots are outliers
```



than Q3

Cap the outliers with Q1, which are having less than Q1

Find the outliers

Q3+1.5*IQR> and Q1-1.5*IQR

Step-1: Calculate Q1 Q2 Q3

Step-2: Calculate IQR=(Q3-Q1)

Step-3: UB=Q3+1.5*IQR

Step-4: LB=Q1-1.5*IQR

Step-5: con1= col>UB

Step-6: con2= col<LB

Step-7: con1|con2

Step-8: col[con1|con2]

-

Removal of outliers

Impute the outliers with medain value

because medain is not impact by Outliers

Cap the outliers with Q3, which are having more

In [13]:

```
#Step-1: Calculate Q1 Q2 Q3
q1=np.quantile(visa_df['prevailing_wage'],0.25)
q2=np.quantile(visa_df['prevailing_wage'],0.50)
q3=np.quantile(visa_df['prevailing_wage'],0.75)

#Step-2:Calculate IQR=(Q3-Q1)
IQR=q3-q1

#Step-3: UB=Q3+1.5*IQR
ub=q3+1.5*IQR

#Step-4: LB=Q1-1.5*IQR
lb=q1-1.5*IQR

#Step-5: con1= col>UB
#Step-6: con2= col<LB

con1=visa_df['prevailing_wage']>ub
con2=visa_df['prevailing_wage']<lb

#step-7 and step-8
outliers=visa_df['prevailing_wage'][con1|con2]
```

```
# series into array of values by applying a .values
outliers_data=outliers.values
len(outliers_data)
```

Out[13]: 427

In [16]:
```
def outliers():

q1=np.quantile(visa_df['prevailing_wage'],0.25)
q2=np.quantile(visa_df['prevailing_wage'],0.50)
q3=np.quantile(visa_df['prevailing_wage'],0.75)  IQR=q3-q1
 ub=q3+1.5*IQR
 lb=q1-1.5*IQR

con1=visa_df['prevailing_wage']>ub
con2=visa_df['prevailing_wage']<lb

##############################
#################
outliers=visa_df['prevailing_wage'][con1|con2]
##############################
#################
outliers_data=outliers.values
 return(outliers_data)

outliers_data=outliers()
len(outliers_data)
```

Out[16]: 427

In [17]:
```
q1=np.quantile(visa_df['prevailing_wage'],0.25)
q2=np.quantile(visa_df['prevailing_wage'],0.50)
q3=np.quantile(visa_df['prevailing_wage'],0.75)
IQR=q3-q1
ub=q3+1.5*IQR
lb=q1-1.5*IQR
con1=visa_df['prevailing_wage']>ub
con2=visa_df['prevailing_wage']<lb
#######################################
########
outliers_df=visa_df[con1|con2] #
outliers dataframe w.r.t p_wage (427)
#######################################
########
outliers_df
```

Out[17]:

	case_id	continent	education_of_employee	has_job_experience	requires_job_traini
14	EZYV15	Asia	Master's	Y	
34	EZYV35	Asia	Master's	N	
130	EZYV131	South America	High School	N	
216	EZYV217	Asia	Master's	Y	
221	EZYV222	North America	Doctorate	Y	
...	...	...	...	...	...

25191 EZYV25192 Asia Master's N

25195 EZYV25196 North America Master's Y

25468 EZYV25469 Asia Bachelor's N

25469 EZYV25470 North America Master's Y

25476 EZYV25477 Asia High School Y

427 rows × 12 columns

```
len(outliers_data),len(visa_df),len(outliers_data)*100/len(visa_df)
```

In [21]:

Out[21]: (427, 25480, 1.6758241758241759)

– 1

:

Removal of outliers

we have 427 outliers in pre_wage column
that means we need to remove 427 rows from entire dataframe

In [23]:

```
q1=np.quantile(visa_df['prevailing_wage'],0.25)
q2=np.quantile(visa_df['prevailing_wage'],0.50)
q3=np.quantile(visa_df['prevailing_wage'],0.75)
IQR=q3-q1
ub=q3+1.5*IQR
lb=q1-1.5*IQR
con1=visa_df['prevailing_wage']<ub
con2=visa_df['prevailing_wage']>lb
############################
###################
non_outliers_df=visa_df[con1&con2]
############################
###################
non_outliers_df
```

Out[23]:

	case_id	continent	education_of_employee	has_job_experience	requires_job_traini
0	EZYV01	Asia	High School	N	
1	EZYV02	Asia	Master's	Y	
2	EZYV03	Asia	Bachelor's	N	
3	EZYV04	Asia	Bachelor's	N	
4	EZYV05	Africa	Master's	Y	
...	...	...	...	...	...

25474 EZYV25475 Africa Doctorate N

25475 EZYV25476 Asia Bachelor's Y

25477 EZYV25478 Asia Master's Y

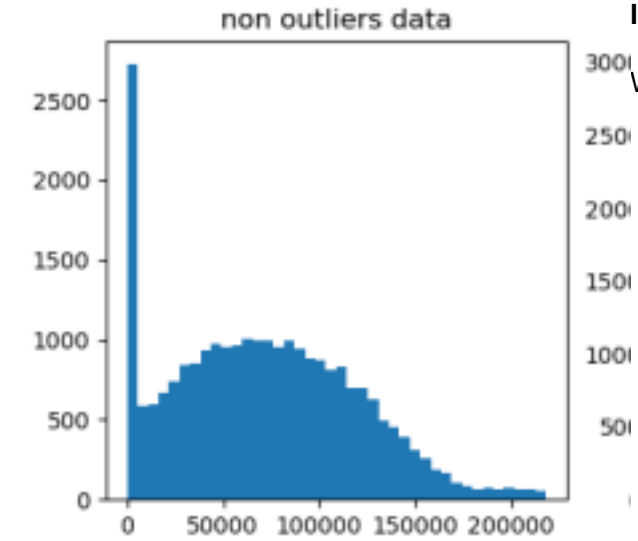
25478 EZYV25479 Asia Master's Y

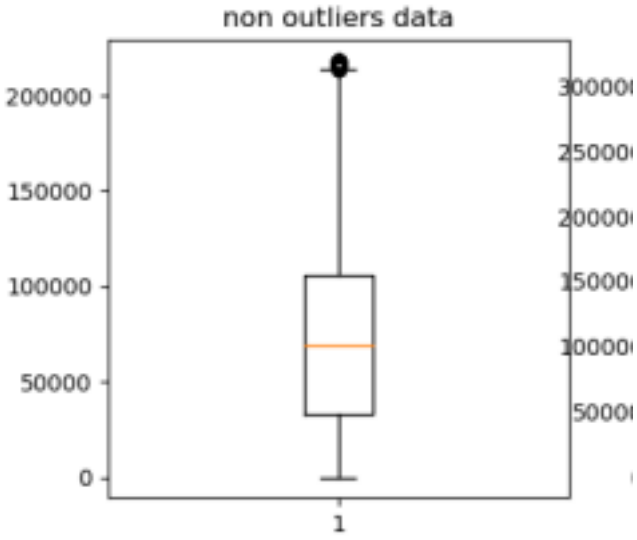
25479 EZYV25480 Asia Bachelor's Y

25053 rows × 12 columns

In [36]:

```
plt.figure(figsize=(8,8))
plt.subplot(2,2,1)
plt.title("non outliers data")
plt.hist(non_outliers_df['prevailing_wage
'],bins=40)

plt.subplot(2,2,2)
plt.title("original data")
plt.hist(visa_df['prevailing_wage'],bins=
40)

plt.subplot(2,2,3)
plt.title("non outliers data")
plt.boxplot(non_outliers_df['prevailing_w
age'])

plt.subplot(2,2,4)
plt.title("original data")
plt.boxplot(visa_df['prevailing_wage'])
plt.show()
```





Impute with Median

We got pre_wage has 427 outliers

l th 427 ith di l f ub,lb

In [39]:

Out[39]: (218315.56125000003, -76564.56875000002)

In [ ]: In [40]:

```
.values:  # if condition:
 # append median
 #else:
 # append i
```

```
592.2029
83425.65
122996.86
83434.03
149907.39
78252.14
53635.39
418.2298
74362.19
67514.76
83588.56
70813.09
28663.05
107196.03
220081.73
74108.02
3706.79
16132.61
150441.13
79948 12
```

In [1]:

```
# iterate through
pre_wages as i
# if a value>ub or <lb
===== > median # else: i
```

```
# Import pacakages
# Read the data

import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot
as plt
```

```

```

```
new_values=[]
for i in
visa_df['prevailing_wage']
```


In [2]:

```
file_path="C:\\Users\\omkar\\OneDrive\\Do
cuments\\Data science\\Naresh IT\\
visa_df=pd.read_csv(file_path)
visa_df
```

Out[2]:

	case_id	continent	education_of_employee	has_job_experience	requires_job_traini
0	EZYV01	Asia	High School	N	
1	EZYV02	Asia	Master's	Y	
2	EZYV03	Asia	Bachelor's	N	
3	EZYV04	Asia	Bachelor's	N	
4	EZYV05	Africa	Master's	Y	

... ...

25475 EZYV25476 Asia Bachelor's Y

25476 EZYV25477 Asia High School Y

25477 EZYV25478 Asia Master's Y

25478 EZYV25479 Asia Master's Y

25479 EZYV25480 Asia Bachelor's Y

25480 rows × 12 columns

In [5]:

```
q1=np.quantile(visa_df['prevailing_wage'],0.25)
q2=np.quantile(visa_df['prevailing_wage'],0.50)
q3=np.quantile(visa_df['prevailing_wage'],0.75)
IQR=q3-q1
ub=q3+1.5*IQR
lb=q1-1.5*IQR

con1=visa_df['prevailing_wage']>ub
con2=visa_df['prevailing_wage']<lb
##########################
###################
outliers=visa_df['prevailing_wage'][con1|con2]
##########################
###################
len(outliers)
```

Out[5]: 427

In [10]:

```
new_data=[]
for i in visa_df['prevailing_wage']:
 if i>ub or i<lb:
    new_data.append(visa_df['prevailing_wage'].median)
 else:
    new_data.append(i)

len(new_data)

# We are iterate trough pre_wage data
# if any datapoint >ub or <lb means it is a outliers so in that postition # we are keeping medain value of the column

# otherwise we are keeping the same value
```

Out[10]: 25480

```
In [13]:
dict1={'Col1':[1,2,3,4],
 'Col2':['A','B','C','D']}

data=pd.DataFrame(dict1)
data

# I want to impute with a value 100 in the col1 #
which are having values >2

# Col1 Col2
# 1 A
# 2 B
# 100 C
# 100 D
```

Out[13]:

	Col1	Col2
0	1	A
1	2	B
2	3	C
3	4	D

np.where will take 3 argument values
Condition : con=data['Col1']>2
If that condition is True will provide the value:100
If that condition is False will keep the same value: data['Col1']
np.where(,,)

```
In [16]:
con=data['Col1']>2  # binary conditions
np.where(con,100,da # True False
ta['Col1'])         # if else

Out[16]: array([  1,  2, 100, 100], dtype=int64)
In [17]:
data

Out[17]:
```

	Col1	Col2
0	1	A
1	2	B
2	3	C
3	4	D

������ – 1

:

Create a new column

In [18]:
```
data['new_col']=[100,200,300,400]
data
```

Out[18]:

	Col1	Col2	new_col
0	1	A	100
1	2	B	200
2	3	C	300
3	4	D	400

In [20]:
```
data['Col3']=np.where(con,100,data['Col1'])
data
con=data['Col1']>2
```

Out[20]:

	Col1	Col2	new_col	Col3
0	1	A	100	1
1	2	B	200	2
2	3	C	300	100
3	4	D	400	100

������ – 2

:

Overwrite the column values

In [21]:
```
data['Col1']=np.where(con,100,data['Col1'])
data
con=data['Col1']>2
```

Out[21]:

	Col1	Col2	new_col	Col3
0	1	A	100	1
1	2	B	200	2
2	100	C	300	100
3	100	D	400	100

In [22]:
```
#Drop unwanted
columns
```

In [23]:
```
data.drop(['new_col
','Col3'],  axis=1,
 inplace=True) data
```

Out[23]:

	Col1	Col2
0	1	A
1	2	B
2	100	C
3	100	D

Implement the same thing for Prevailaing wage

In [ ]:

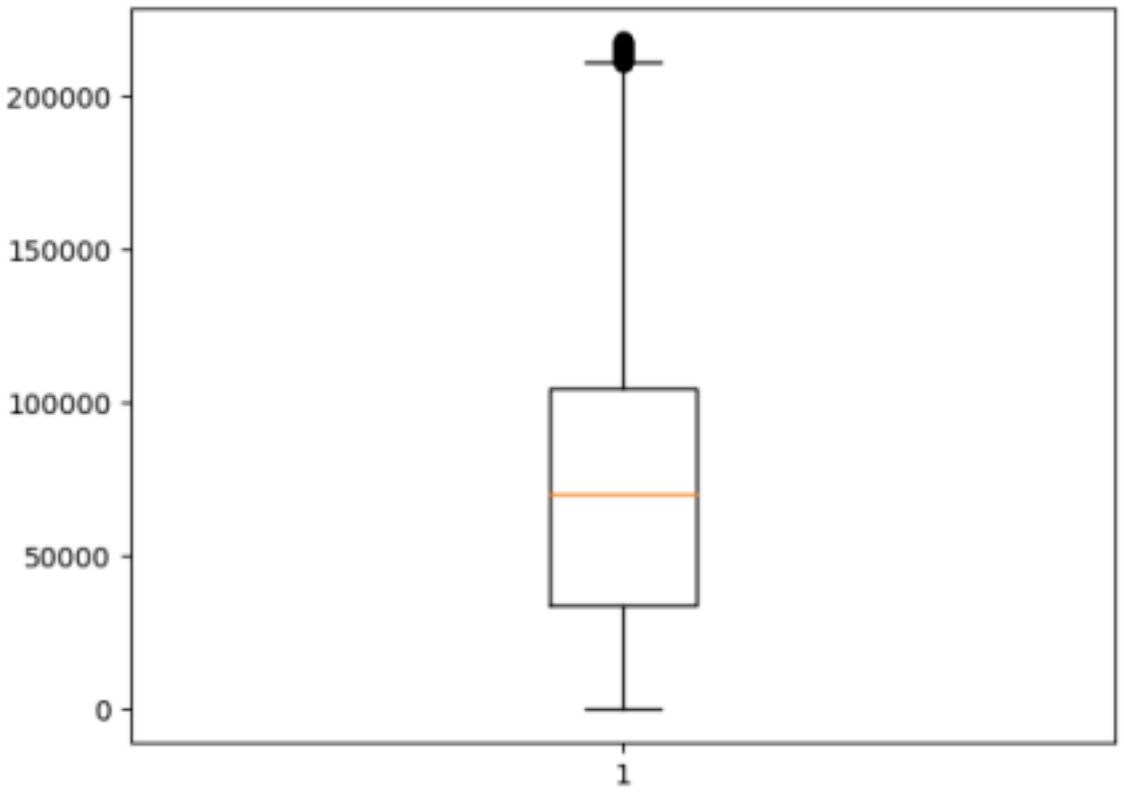
In [25]:
```
con1=visa_df['prevailing_wage']>ub
con2=visa_df['prevailing_wage']<lb
con=con1|con2
wage_median=visa_df['prevailing_wage'].med
ian()
visa_df['prevailing_wage']=np.where(con,
 wage_median,
```

```
# step-1: write the condition
# step-2: True value: Medain value
# Step-3: False value: same column values
# Step-4: implment
np.where(<con1>,<True_vale>,<False_vale>)
# Step-5: Overwrite in the same column
name
# Step-6: Draw the boxplot for p_Wage
# Step-7: Daraw the histogram p_wage


###################### Read the data
###################################
file_path="C:\\Users\\omkar\\OneDrive\\Doc
uments\\Data science\\Naresh IT\\
visa_df=pd.read_csv(file_path)

                                        visa_df['prevailing_wage'])
##########################np.where#########
###################################
```


In [26]:
```
plt.boxplot(visa_df['pre
vailing_wage'])
```

Out[26]:
```
{'whiskers': [<matplotlib.lines.Line2D at 0x20d76418fd0>,
  <matplotlib.lines.Line2D at 0x20d7641d350>],
 'caps': [<matplotlib.lines.Line2D at 0x20d763a0d10>,
  <matplotlib.lines.Line2D at 0x20d76423d10>],
```

```
'boxes': [<matplotlib.lines.Line2D at 0x20d7640ff90>],
'medians': [<matplotlib.lines.Line2D at 0x20d7642b790>],
'fliers': [<matplotlib.lines.Line2D at 0x20d766393d0>],
'means': []}
```



In [ ]: