In [60]:
```
age=[24,25,30,19,20,24,20,20,32,20,18,46]
```

In [2]:
```
import numpy as np
```

In [3]:
```
np.mean(age)
```

Out[3]:
```
24.833333333333332
```

In [4]:
```
np.median(age)
```

Out[4]:
```
22.0
```

In [5]:
```
from statistics import mode
```

In [6]:
```
mode(age)
```

Out[6]:
```
20
```

In [7]:
```
import pandas as pd
```

In [8]:
```
pwd
```

Out[8]:
```
'C:\\Users\\RISHI'
```

In [9]:
```
df=pd.read_excel(r"C:\Users\RISHI\Desktop\module 4\Batch-8\Batch-9\
PurchaseOrders.xlsx")
```

In [10]:
```
df.head(7)
```

Out[10]:

	Supplier	Order No.	Item No.	Item Description	Item Cost	Quantity	Cost per order	A/P Terms (Months)	Order Date	Arrival Date
0	Hulkey Fasteners	Aug11001	1122	Airframe fasteners	4.25	19500	82875.0	30	2011-08-05	2011-08-13
1	Alum Sheeting	Aug11002	1243	Airframe fasteners	4.25	10000	42500.0	30	2011-08-08	2011-08-14
2	Fast-Tie Aerospace	Aug11003	5462	Shielded Cable/ft.	1.05	23000	24150.0	30	2011-08-10	2011-08-15
3	Fast-Tie Aerospace	Aug11004	5462	Shielded Cable/ft.	1.05	21500	22575.0	30	2011-08-15	2011-08-22
4	Steelpin Inc.	Aug11005	5319	Shielded Cable/ft.	1.10	17500	19250.0	30	2011-08-20	2011-08-31
5	Fast-Tie Aerospace	Aug11006	5462	Shielded Cable/ft.	1.05	22500	23625.0	30	2011-08-20	2011-08-26
6	Steelpin Inc.	Aug11007	4312	Bolt-nut package	3.75	4250	15937.5	30	2011-08-25	2011-09-01

In [11]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 94 entries, 0 to 93
Data columns (total 10 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   Supplier            94 non-null     object
 1   Order No.           94 non-null     object
 2   Item No.            94 non-null     int64
 3   Item Description    94 non-null     object
 4   Item Cost           94 non-null     float64
 5   Quantity            94 non-null     int64
 6   Cost per order      94 non-null     float64
 7   A/P Terms (Months)  94 non-null     int64
 8   Order Date          94 non-null     datetime64[ns]
 9   Arrival Date        94 non-null     datetime64[ns]
dtypes: datetime64[ns](2), float64(2), int64(3), object(3)
memory usage: 7.5+ KB
```

In [12]:
```
df.describe()
```

Out[12]:

	Item No.	Item Cost	Quantity	Cost per order	A/P Terms (Months)	Order Date	
count	94.000000	94.000000	94.000000	94.000000	94.000000	94	
mean	5616.553191	62.640957	5857.404255	26295.319149	30.638298	2011-09-26 15:19:08.936170240	01:16:3
min	1122.000000	0.550000	90.000000	68.750000	15.000000	2011-08-05 00:00:00	
25%	4139.250000	1.850000	500.000000	6757.812500	30.000000	2011-09-05 00:00:00	
50%	5462.000000	3.700000	1915.000000	15656.250000	30.000000	2011-09-28 12:00:00	
75%	7258.000000	4.250000	9750.000000	27593.750000	30.000000	2011-10-15 00:00:00	
max	9977.000000	655.500000	25000.000000	127500.000000	45.000000	2011-11-05 00:00:00	
std	2593.745797	142.004510	7252.403704	29842.831197	6.808993	NaN	

In [22]:
```
df[['Quantity']].mean()
```

Out[22]:
```
Quantity    5857.404255
dtype: float64
```

In [18]:
```
df['Quantity'].median()
```

Out[18]:
```
1915.0
```

In [19]:
```
df['Quantity'].mode()
```

Out[19]:
```
0    150
Name: Quantity, dtype: int64
```

In [20]:
```
from statistics import mode
```

In [21]:
```
mode(df['Quantity'])
```

Out[21]:
```
150
```

In [23]:
```
np.std(age)
```

Out[23]:
```
7.646713164636308
```

In [24]:
```
np.var(age)
```

Out[24]:
```
58.47222222222222
```

In [25]:
```
np.max(age)
```

Out[25]:
```
46
```

In [26]:
```
np.min(age)
```

Out[26]:
```
18
```

In [28]:
```
df['Quantity'].std()
```

Out[28]:
```
7252.403704340681
```

In [29]:
```
df['Quantity'].var()
```

Out[29]:
```
52597359.49073443
```

In [34]:
```
cv_quantity=(df['Quantity'].std()/df['Quantity'].mean())*100
```

In [35]:
```
cv_quantity
```

Out[35]:
```
123.81600088050475
```

In [37]:
```
df['Quantity'].quantile()
```

Out[37]:
```
1915.0
```

In [38]:
```
df['Quantity'].quantile(0.5)
```

Out[38]:
```
1915.0
```

In [39]:
```
df['Quantity'].quantile([.25, .5, .75])
```

Out[39]:
```
0.25     500.0
0.50    1915.0
0.75    9750.0
Name: Quantity, dtype: float64
```

In [42]:
```
df['Quantity'].skew()
```

Out[42]:
```
1.2556193839396843
```

In [43]:
```
df['Quantity'].kurtosis()
```

Out[43]:
```
0.2585630882738843
```

In [45]:
```
name=['Abhilasha','Ashwini','Mandeep', 'Prataksh']
```

In [46]:
```
age=[22,21,30,25]
```

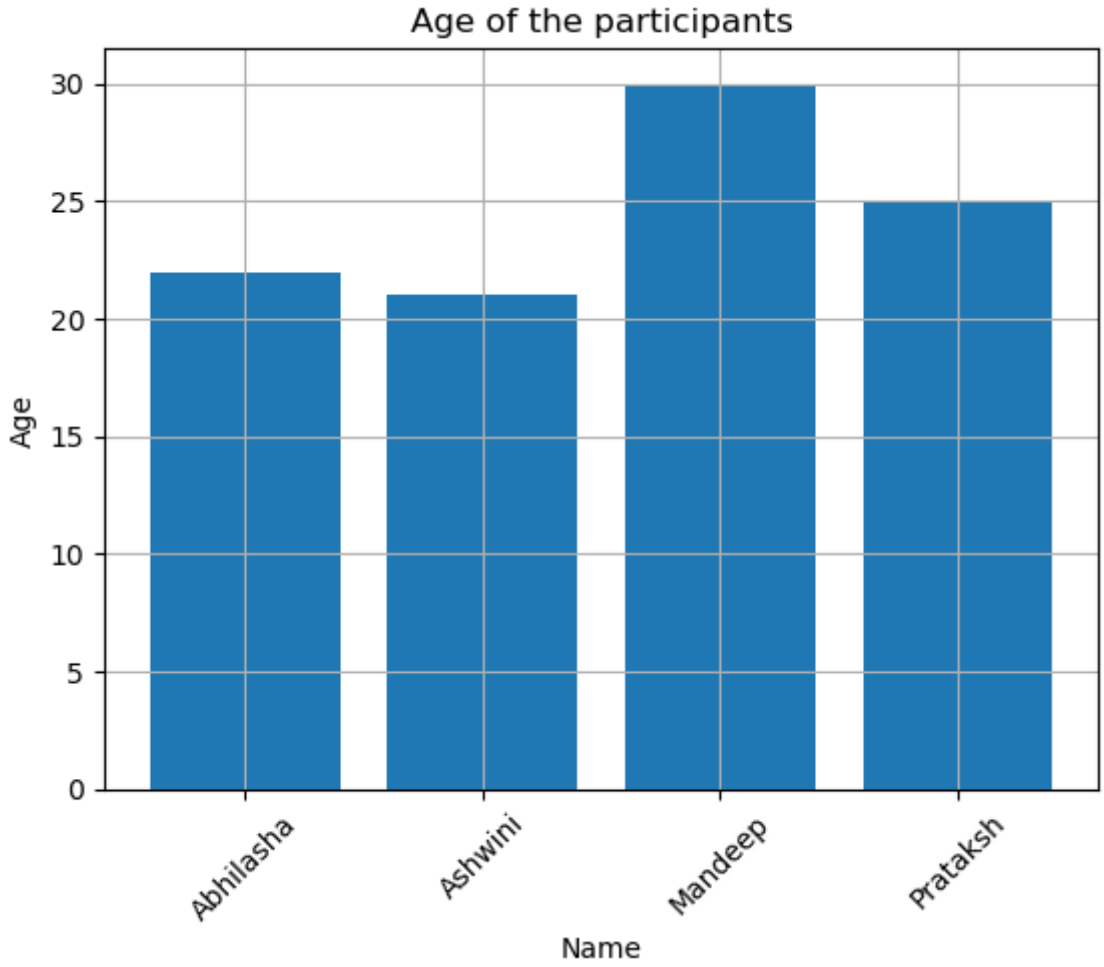
In [47]:
```
import matplotlib.pyplot as plt
```

In [48]:
```
import seaborn as sns
```

```
C:\Users\RISHI\anaconda3\lib\site-packages\scipy\__init__.py:155: UserWarning: A N
umPy version >=1.18.5 and <1.25.0 is required for this version of SciPy (detected
version 1.26.1
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion}"
```

In [55]:
```
plt.bar(name,age)
plt.xlabel("Name")
plt.ylabel("Age")
plt.title("Age of the participants")
plt.grid()
plt.xticks(rotation=45)
```

Out[55]:
```
([0, 1, 2, 3],
 [Text(0, 0, ''), Text(0, 0, ''), Text(0, 0, ''), Text(0, 0, '')])
```

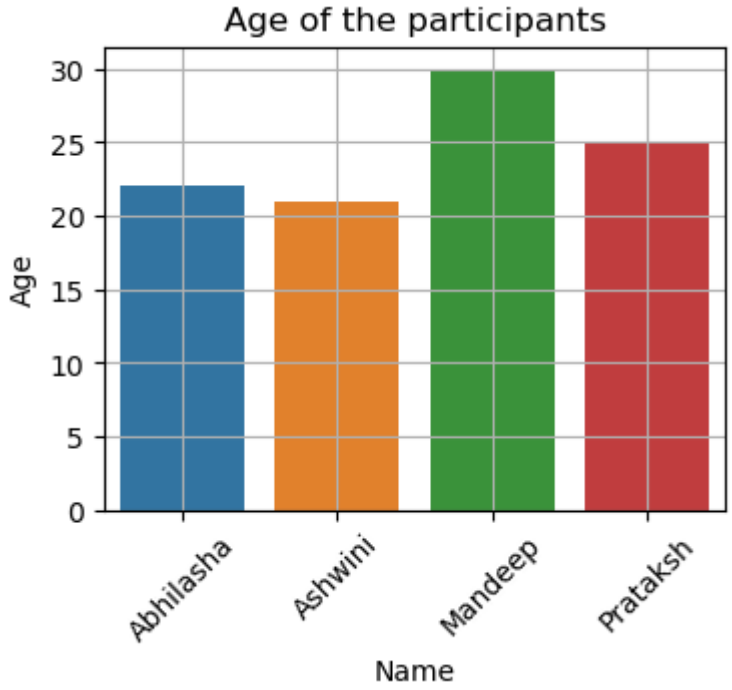


In [57]:
```
plt.figure(figsize=(4,3))
sns.barplot(x=name,y=age)
plt.xlabel("Name")
plt.ylabel("Age")
plt.title("Age of the participants")
plt.grid()
plt.xticks(rotation=45)
```

```
C:\Users\RISHI\anaconda3\lib\site-packages\seaborn\_core.py:1225: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\RISHI\anaconda3\lib\site-packages\seaborn\_core.py:1225: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\RISHI\anaconda3\lib\site-packages\seaborn\_core.py:1485: FutureWarning: unique with argument that is not not a Series, Index, ExtensionArray, or np.ndarray is deprecated and will raise in a future version.
  order = pd.unique(vector)
C:\Users\RISHI\anaconda3\lib\site-packages\seaborn\_core.py:1225: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
```

Out[57]:

```
(array([0, 1, 2, 3]),
 [Text(0, 0, 'Abhilasha'),
  Text(1, 0, 'Ashwini'),
  Text(2, 0, 'Mandeep'),
  Text(3, 0, 'Prataksh')])
```



In [58]:

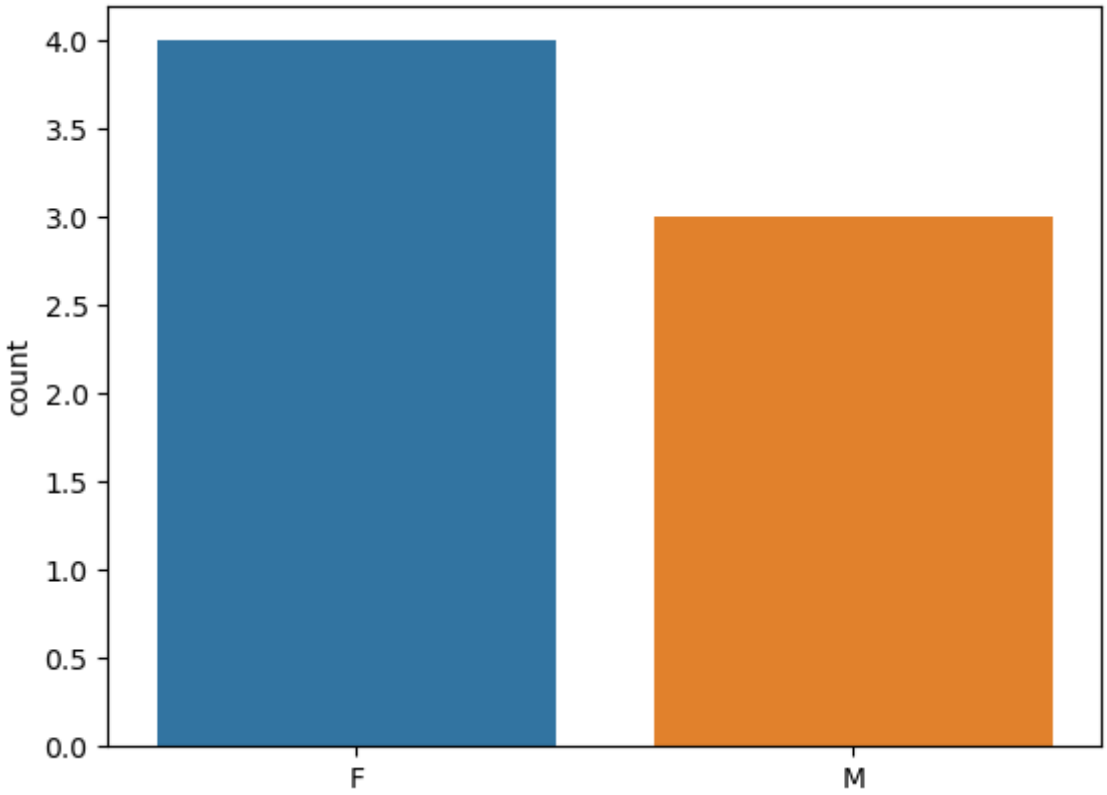
```
Gender=["F","F","F","M","M","M","F"]
```

In [59]:

```
sns.countplot(x=Gender)
```

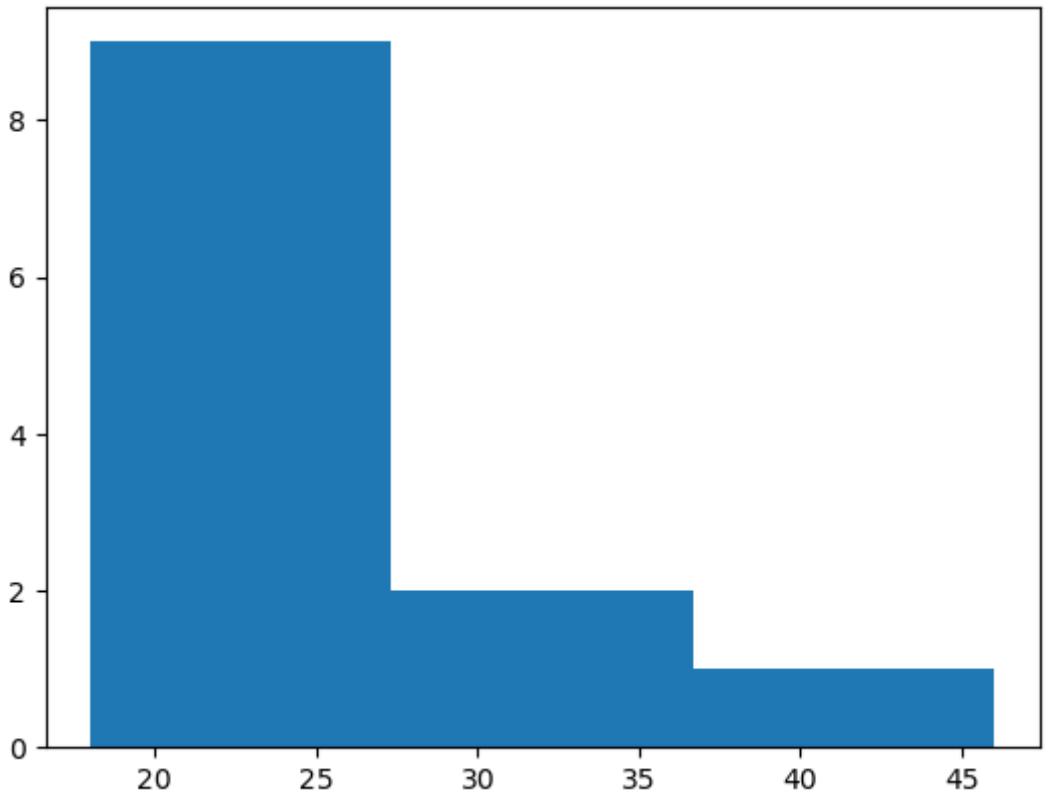
```
C:\Users\RISHI\anaconda3\lib\site-packages\seaborn\_core.py:1225: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\RISHI\anaconda3\lib\site-packages\seaborn\_core.py:1225: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\RISHI\anaconda3\lib\site-packages\seaborn\_core.py:1485: FutureWarning: unique with argument that is not not a Series, Index, ExtensionArray, or np.ndarray is deprecated and will raise in a future version.
  order = pd.unique(vector)
C:\Users\RISHI\anaconda3\lib\site-packages\seaborn\_core.py:1225: FutureWarning: is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
```

Out[59]:
```
<AxesSubplot:ylabel='count'>
```



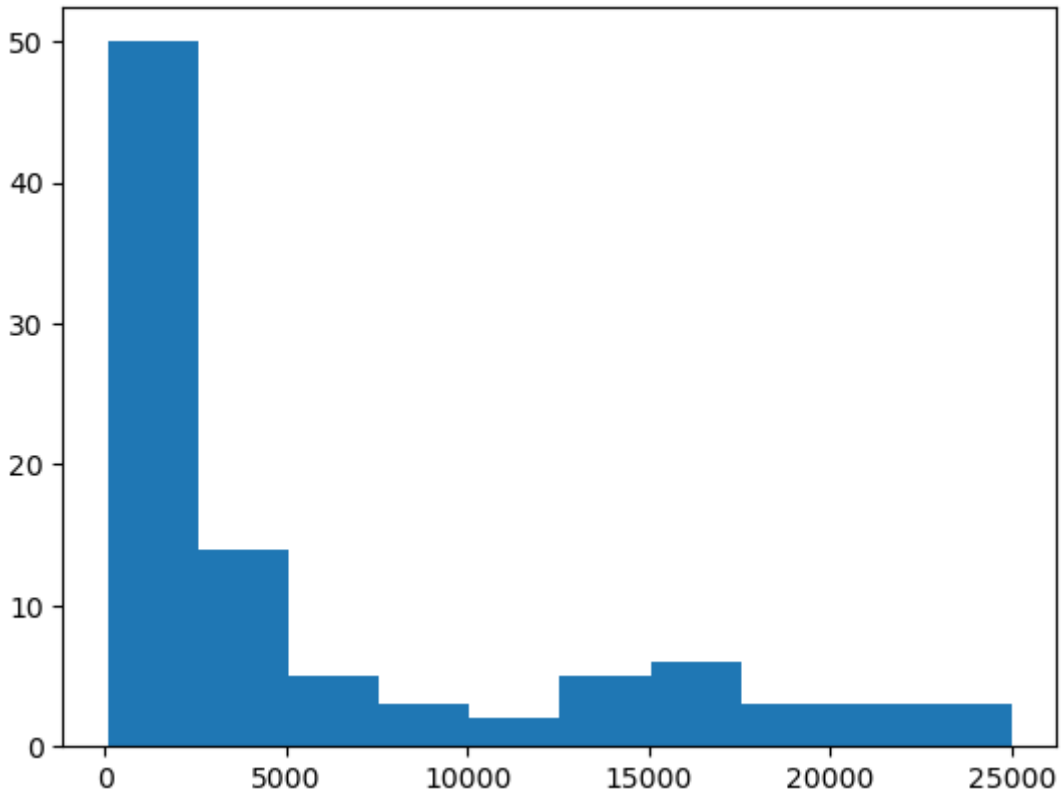
In [62]:
```
plt.hist(age, bins=3)
```

Out[62]:
```
(array([9., 2., 1.]),
 array([18.        , 27.33333333, 36.66666667, 46.        ]),
 <BarContainer object of 3 artists>)
```



In [63]:
```
plt.hist(df['Quantity'])
```

Out[63]:
```
(array([50., 14.,  5.,  3.,  2.,  5.,  6.,  3.,  3.,  3.]),
 array([   90.,  2581.,  5072.,  7563., 10054., 12545., 15036., 17527.,
        20018., 22509., 25000.]),
 <BarContainer object of 10 artists>)
```

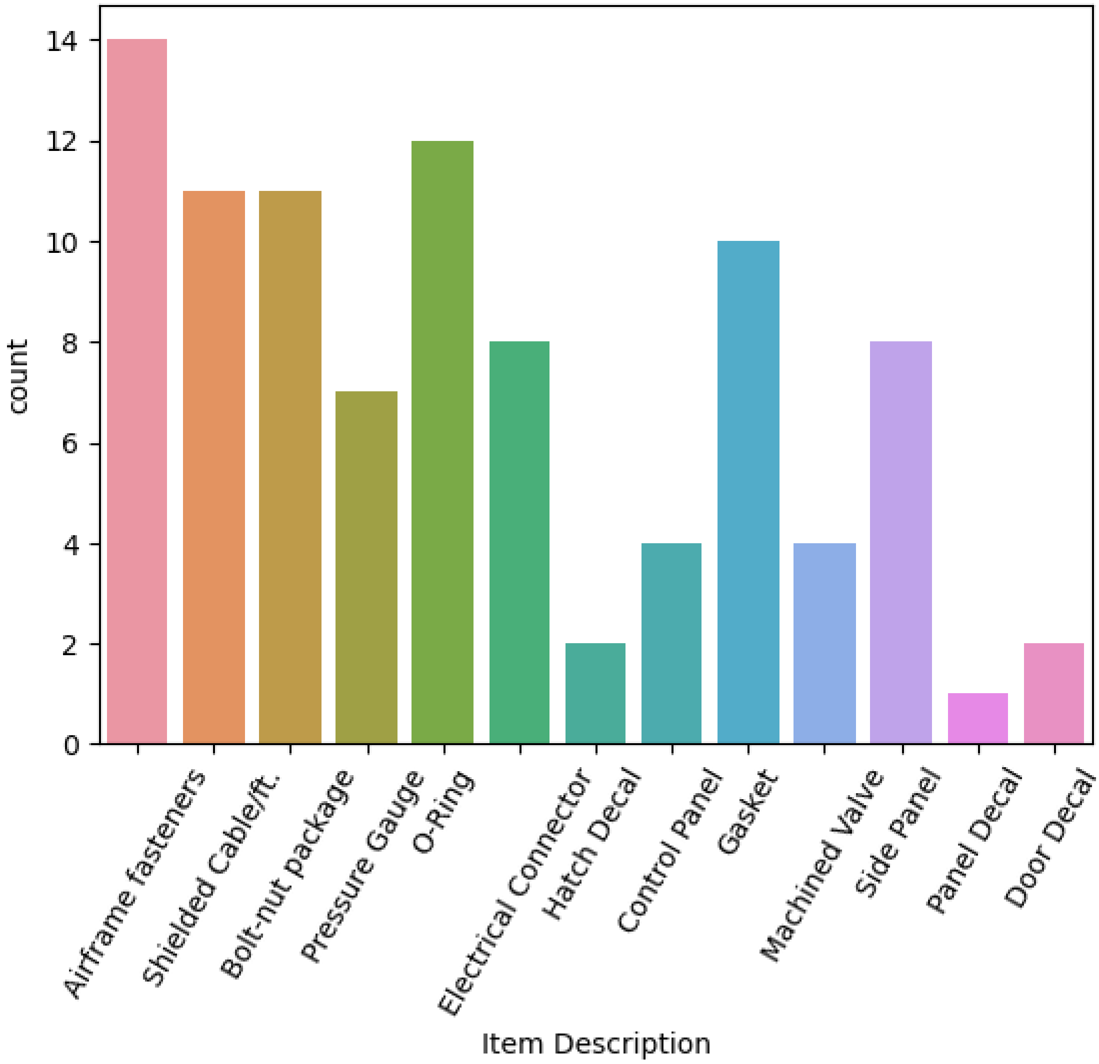


In [67]:
```
sns.countplot(x='Item Description', data=df)
plt.xticks(rotation=60)
```

```
C:\Users\RISHI\anaconda3\lib\site-packages\seaborn\_core.py:1225: FutureWarning: i
s_categorical_dtype is deprecated and will be removed in a future version. Use isi
nstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\RISHI\anaconda3\lib\site-packages\seaborn\_core.py:1225: FutureWarning: i
s_categorical_dtype is deprecated and will be removed in a future version. Use isi
nstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\RISHI\anaconda3\lib\site-packages\seaborn\_core.py:1225: FutureWarning: i
s_categorical_dtype is deprecated and will be removed in a future version. Use isi
nstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
```

Out[67]:

```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12]),
 [Text(0, 0, 'Airframe fasteners'),
  Text(1, 0, 'Shielded Cable/ft.'),
  Text(2, 0, 'Bolt-nut package'),
  Text(3, 0, 'Pressure Gauge'),
  Text(4, 0, 'O-Ring'),
  Text(5, 0, 'Electrical Connector'),
  Text(6, 0, 'Hatch Decal'),
  Text(7, 0, 'Control Panel'),
  Text(8, 0, 'Gasket'),
  Text(9, 0, 'Machined Valve'),
  Text(10, 0, 'Side Panel'),
  Text(11, 0, 'Panel Decal'),
  Text(12, 0, 'Door Decal')])
```



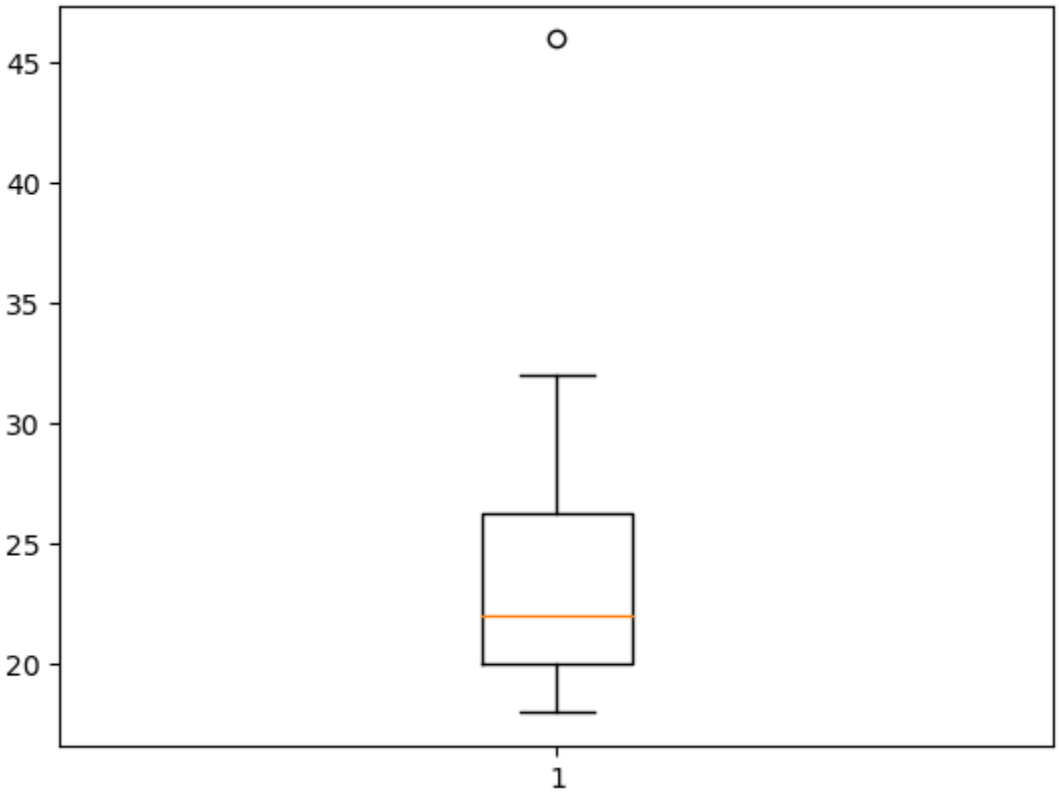
In [72]:

```
age
```

Out[72]:
```
[24, 25, 30, 19, 20, 24, 20, 20, 32, 20, 18, 46]
```

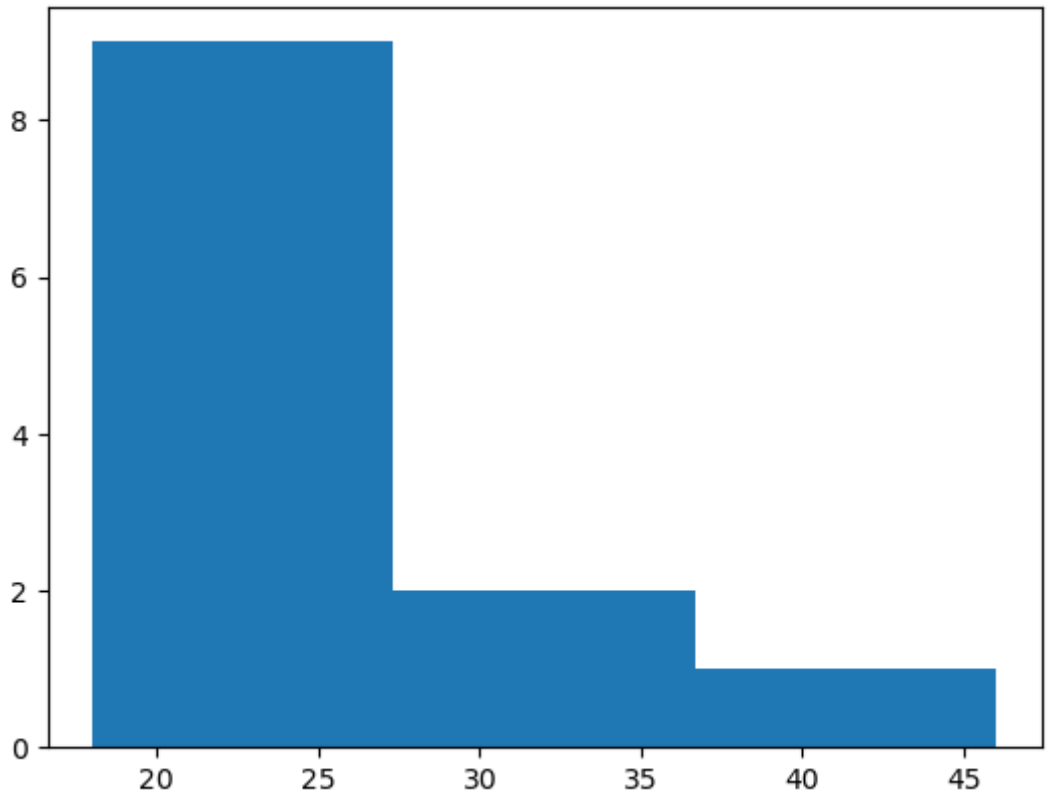
In [73]:
```
plt.boxplot(age)
```

Out[73]:
```
{'whiskers': [<matplotlib.lines.Line2D at 0x179c9d2cc40>,
  <matplotlib.lines.Line2D at 0x179c9d2cf10>],
 'caps': [<matplotlib.lines.Line2D at 0x179c9d3c250>,
  <matplotlib.lines.Line2D at 0x179c9d3c520>],
 'boxes': [<matplotlib.lines.Line2D at 0x179c9d2c970>],
 'medians': [<matplotlib.lines.Line2D at 0x179c9d3c7f0>],
 'fliers': [<matplotlib.lines.Line2D at 0x179c9d3cac0>],
 'means': []}
```



In [75]:
```
plt.hist(age,bins=3)
```

Out[75]:
```
(array([9., 2., 1.]),
 array([18.        , 27.33333333, 36.66666667, 46.        ]),
 <BarContainer object of 3 artists>)
```



In [76]:
```
height=[155,167,160,180,174,175]
weight=[56,70,78, 80,75,68]
```

In [77]:
```
arr1=np.array(height)
```

In [78]:
```
arr2=np.array(weight)
```

In [80]:
```
np.corrcoef(height,weight)[0,1]
```

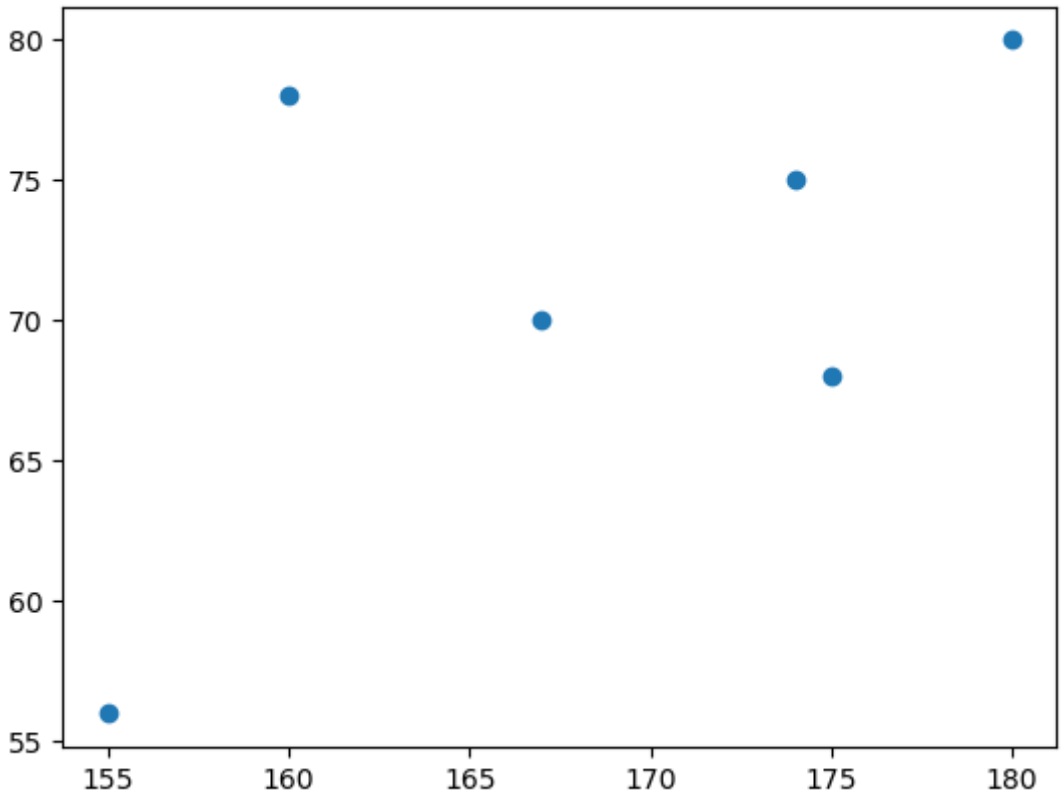
Out[80]:
```
0.5975226885094456
```

In [81]:
```
df['Item Cost'].corr(df['Quantity'])
```

Out[81]:
```
-0.3342939438535476
```

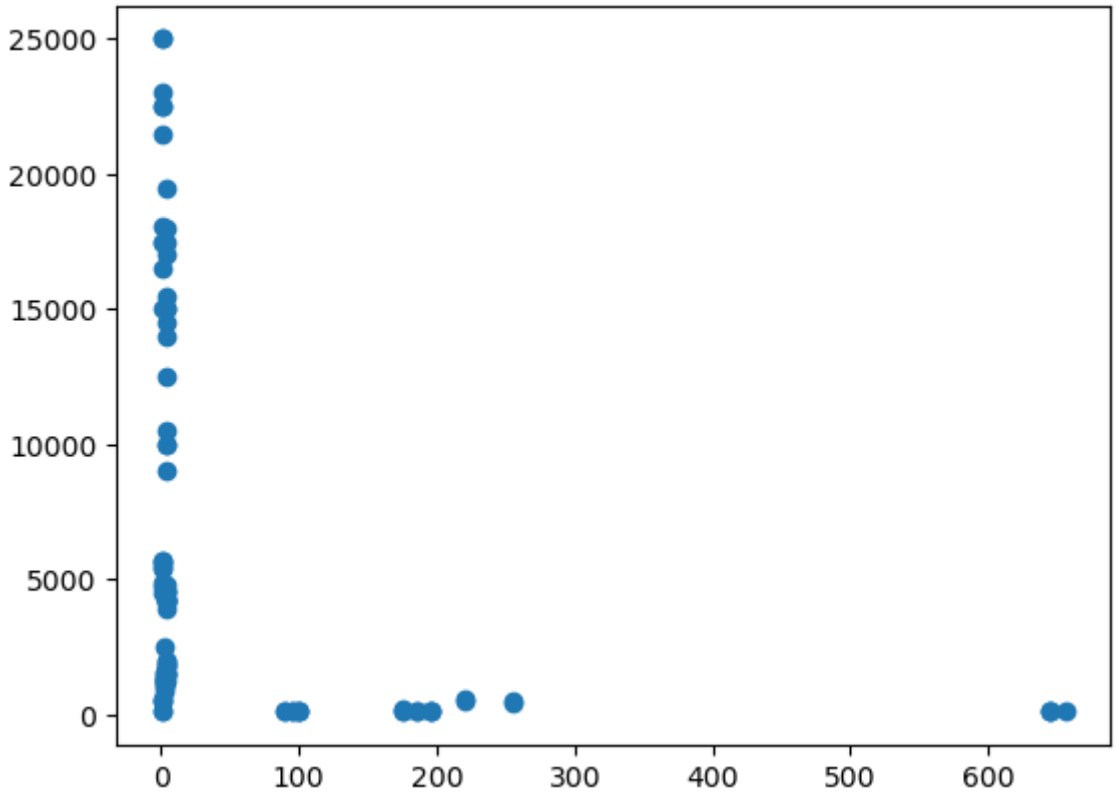
In [82]:
```
plt.scatter(height,weight)
```

Out[82]:
```
<matplotlib.collections.PathCollection at 0x179ca219370>
```



In [83]:
```
plt.scatter(df['Item Cost'], df['Quantity'])
```

Out[83]:
```
<matplotlib.collections.PathCollection at 0x179cb24c460>
```



In [87]:
```
gender=['M','F','F','M','M', 'M']
```

In [88]:
```
height
```

Out[88]:
```
[155, 167, 160, 180, 174, 175]
```

In [89]:

```
df1=pd.DataFrame({'Height':height, 'Weight':weight, 'Gender': gender})
df1
```

Out[89]:

	Height	Weight	Gender
0	155	56	M
1	167	70	F
2	160	78	F
3	180	80	M
4	174	75	M
5	175	68	M

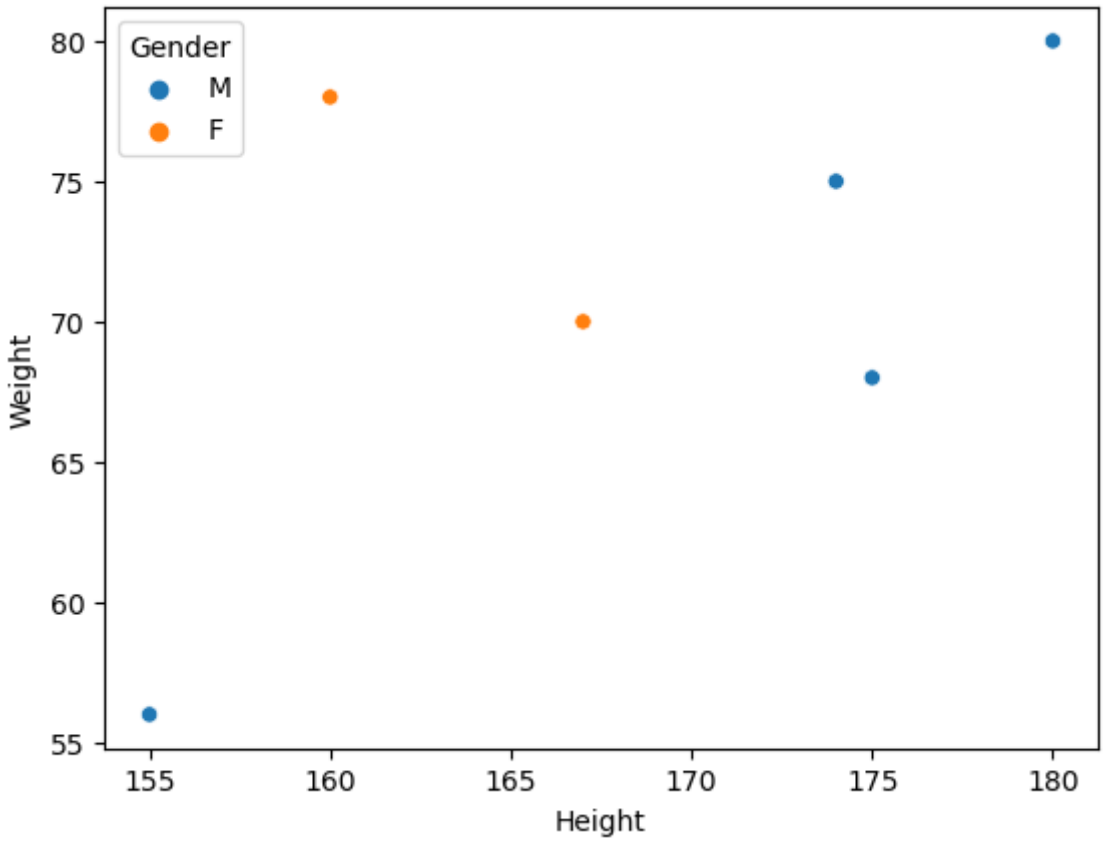
In [91]:

```
sns.scatterplot(x='Height',y='Weight', data=df1, hue='Gender')
```

```
C:\Users\RISHI\anaconda3\lib\site-packages\seaborn\_core.py:1225: FutureWarning: i
s_categorical_dtype is deprecated and will be removed in a future version. Use isi
nstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\RISHI\anaconda3\lib\site-packages\seaborn\_core.py:1225: FutureWarning: i
s_categorical_dtype is deprecated and will be removed in a future version. Use isi
nstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\RISHI\anaconda3\lib\site-packages\seaborn\_core.py:1225: FutureWarning: i
s_categorical_dtype is deprecated and will be removed in a future version. Use isi
nstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\RISHI\anaconda3\lib\site-packages\seaborn\_core.py:1225: FutureWarning: i
s_categorical_dtype is deprecated and will be removed in a future version. Use isi
nstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\RISHI\anaconda3\lib\site-packages\seaborn\_core.py:1225: FutureWarning: i
s_categorical_dtype is deprecated and will be removed in a future version. Use isi
nstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
```

Out[91]:

```
<AxesSubplot:xlabel='Height', ylabel='Weight'>
```


Gender
M
F
80
75
70
65
60
55
Weight
155
160
165
170
175
180
Height

In [ ]: