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

In [2]:
```
data = pd.read_excel(io='https://github.com/insaid2018/Term-1/blob/master/Data/Ca
print('Shape of the dataset:', data.shape)
data.head()
```

```
Shape of the dataset: (541909, 8)
```

Out[2]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom

In [3]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 8 columns):
 #   Column       Non-Null Count   Dtype
---  ------       --------------   -----
 0   InvoiceNo    541909 non-null  object
 1   StockCode    541909 non-null  object
 2   Description  540455 non-null  object
 3   Quantity     541909 non-null  int64
 4   InvoiceDate  541909 non-null  datetime64[ns]
 5   UnitPrice    541909 non-null  float64
 6   CustomerID   406829 non-null  float64
 7   Country      541909 non-null  object
dtypes: datetime64[ns](1), float64(2), int64(1), object(4)
memory usage: 33.1+ MB
```

In [5]:
```
data.to_csv('sandeep_pandas')
```

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

In [10]:
```
data= pd.read_csv('sandeep_pandas' Index = False)
```

```
  File "C:\Users\amala\AppData\Local\Temp/ipykernel_37660/327449208.py", line 1
    data= pd.read_csv('sandeep_pandas' Index = False)
                                       ^
SyntaxError: invalid syntax
```

In [4]:
```
data
```

Out[4]:

	Unnamed: 0	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	Custome
0	0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	1785
1	1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	1785
2	2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	1785
3	3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	1785
4	4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	1785
...	...	...	...	...	...	...	...	
541904	541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	2011-12-09 12:50:00	0.85	1268
541905	541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	2011-12-09 12:50:00	2.10	1268
541906	541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	2011-12-09 12:50:00	4.15	1268
541907	541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	2011-12-09 12:50:00	4.15	1268
541908	541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	2011-12-09 12:50:00	4.95	1268

541909 rows × 9 columns

In [5]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 9 columns):
 #   Column       Non-Null Count   Dtype  
---  ------       --------------   -----  
 0   Unnamed: 0   541909 non-null  int64  
 1   InvoiceNo    541909 non-null  object 
 2   StockCode    541909 non-null  object 
 3   Description  540455 non-null  object 
 4   Quantity     541909 non-null  int64  
 5   InvoiceDate  541909 non-null  object 
 6   UnitPrice    541909 non-null  float64
 7   CustomerID   406829 non-null  float64
 8   Country      541909 non-null  object 
dtypes: float64(2), int64(2), object(5)
memory usage: 37.2+ MB
```

In [7]:
```
data.shape
```

Out[7]:
```
(541909, 9)
```

In [9]:
```
data.columns
```

Out[9]:
```
Index(['Unnamed: 0', 'InvoiceNo', 'StockCode', 'Description', 'Quantity',
       'InvoiceDate', 'UnitPrice', 'CustomerID', 'Country'],
      dtype='object')
```

In [14]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 9 columns):
 #   Column       Non-Null Count   Dtype  
---  ------       --------------   -----  
 0   Unnamed: 0   541909 non-null  int64  
 1   InvoiceNo    541909 non-null  object 
 2   StockCode    541909 non-null  object 
 3   Description  540455 non-null  object 
 4   Quantity     541909 non-null  int64  
 5   InvoiceDate  541909 non-null  object 
 6   UnitPrice    541909 non-null  float64
 7   CustomerID   406829 non-null  float64
 8   Country      541909 non-null  object 
dtypes: float64(2), int64(2), object(5)
memory usage: 37.2+ MB
```

In [16]:
```
data.describe()
```

Out[16]:

	Unnamed: 0	Quantity	UnitPrice	CustomerID
count	541909.00000	541909.000000	541909.000000	406829.000000
mean	270954.00000	9.552250	4.611114	15287.690570
std	156435.79785	218.081158	96.759853	1713.600303
min	0.00000	-80995.000000	-11062.060000	12346.000000
25%	135477.00000	1.000000	1.250000	13953.000000
50%	270954.00000	3.000000	2.080000	15152.000000
75%	406431.00000	10.000000	4.130000	16791.000000
max	541908.00000	80995.000000	38970.000000	18287.000000

In [17]:
```
data.drop('Description', inplace = True, axis =1)
```

In [18]:
```
data
```

Out[18]:

	Unnamed: 0	InvoiceNo	StockCode	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	0	536365	85123A	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	1	536365	71053	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	2	536365	84406B	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	3	536365	84029G	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	4	536365	84029E	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
...	...	...	...	...	...	...	...	...
541904	541904	581587	22613	12	2011-12-09 12:50:00	0.85	12680.0	France
541905	541905	581587	22899	6	2011-12-09 12:50:00	2.10	12680.0	France
541906	541906	581587	23254	4	2011-12-09 12:50:00	4.15	12680.0	France
541907	541907	581587	23255	4	2011-12-09 12:50:00	4.15	12680.0	France
541908	541908	581587	22138	3	2011-12-09 12:50:00	4.95	12680.0	France

541909 rows × 8 columns

In [20]:
```
data.reset_index(drop=True)
```

Out[20]:

	Unnamed: 0	InvoiceNo	StockCode	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	0	536365	85123A	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	1	536365	71053	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	2	536365	84406B	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	3	536365	84029G	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	4	536365	84029E	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
...	...	...	...	...	...	...	...	...
541904	541904	581587	22613	12	2011-12-09 12:50:00	0.85	12680.0	France
541905	541905	581587	22899	6	2011-12-09 12:50:00	2.10	12680.0	France
541906	541906	581587	23254	4	2011-12-09 12:50:00	4.15	12680.0	France
541907	541907	581587	23255	4	2011-12-09 12:50:00	4.15	12680.0	France
541908	541908	581587	22138	3	2011-12-09 12:50:00	4.95	12680.0	France

541909 rows × 8 columns

In [21]:
```
data2=data.reset_index(drop=True)
```

In [22]:
```
data2
```

Out[22]:

	Unnamed: 0	InvoiceNo	StockCode	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	0	536365	85123A	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	1	536365	71053	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	2	536365	84406B	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	3	536365	84029G	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	4	536365	84029E	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
...	...	...	...	...	...	...	...	...
541904	541904	581587	22613	12	2011-12-09 12:50:00	0.85	12680.0	France
541905	541905	581587	22899	6	2011-12-09 12:50:00	2.10	12680.0	France
541906	541906	581587	23254	4	2011-12-09 12:50:00	4.15	12680.0	France
541907	541907	581587	23255	4	2011-12-09 12:50:00	4.15	12680.0	France
541908	541908	581587	22138	3	2011-12-09 12:50:00	4.95	12680.0	France

541909 rows × 8 columns

In [23]:
```
data.head()
```

Out[23]:

	Unnamed: 0	InvoiceNo	StockCode	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	0	536365	85123A	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	1	536365	71053	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	2	536365	84406B	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	3	536365	84029G	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	4	536365	84029E	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom

In [24]:
```
data['CustomerID'] = data['CustomerID'].fillna(0)
```

In [25]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 8 columns):
 #   Column       Non-Null Count   Dtype  
---  ------       --------------   -----  
 0   Unnamed: 0   541909 non-null  int64  
 1   InvoiceNo    541909 non-null  object 
 2   StockCode    541909 non-null  object 
 3   Quantity     541909 non-null  int64  
 4   InvoiceDate  541909 non-null  object 
 5   UnitPrice    541909 non-null  float64
 6   CustomerID   541909 non-null  float64
 7   Country      541909 non-null  object 
dtypes: float64(2), int64(2), object(4)
memory usage: 33.1+ MB
```

In [26]:
```
data['CustomerID'] = data['CustomerID'].astype(int).astype('str')
```

In [27]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 8 columns):
 #   Column       Non-Null Count   Dtype  
---  ------       --------------   -----  
 0   Unnamed: 0   541909 non-null  int64  
 1   InvoiceNo    541909 non-null  object 
 2   StockCode    541909 non-null  object 
 3   Quantity     541909 non-null  int64  
 4   InvoiceDate  541909 non-null  object 
 5   UnitPrice    541909 non-null  float64
 6   CustomerID   541909 non-null  object 
 7   Country      541909 non-null  object 
dtypes: float64(1), int64(2), object(5)
memory usage: 33.1+ MB
```

In [28]:
```
data['CustomerID'] = data['CustomerID'].astype(float)
```

In [29]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 8 columns):
 #   Column       Non-Null Count   Dtype  
---  ------       --------------   -----  
 0   Unnamed: 0   541909 non-null  int64  
 1   InvoiceNo    541909 non-null  object 
 2   StockCode    541909 non-null  object 
 3   Quantity     541909 non-null  int64  
 4   InvoiceDate  541909 non-null  object 
 5   UnitPrice    541909 non-null  float64
 6   CustomerID   541909 non-null  float64
 7   Country      541909 non-null  object 
dtypes: float64(2), int64(2), object(4)
memory usage: 33.1+ MB
```

In [30]:
```
data['CustomerID'] = data['CustomerID'].astype(str)
```

In [31]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 8 columns):
 #   Column       Non-Null Count   Dtype  
---  ------       --------------   -----  
 0   Unnamed: 0   541909 non-null  int64  
 1   InvoiceNo    541909 non-null  object 
 2   StockCode    541909 non-null  object 
 3   Quantity     541909 non-null  int64  
 4   InvoiceDate  541909 non-null  object 
 5   UnitPrice    541909 non-null  float64
 6   CustomerID   541909 non-null  object 
 7   Country      541909 non-null  object 
dtypes: float64(1), int64(2), object(5)
memory usage: 33.1+ MB
```

In [32]:
```
data['CustomerID'].unique()
```

Out[32]:
```
array(['17850.0', '13047.0', '12583.0', ..., '13298.0', '14569.0',
       '12713.0'], dtype=object)
```

In [36]:
```
data['CustomerID'].describe()
```

Out[36]:
```
count     541909
unique      4373
top          0.0
freq      135080
Name: CustomerID, dtype: object
```

In [37]:
```
data.describe()
```

Out[37]:

	Unnamed: 0	Quantity	UnitPrice
count	541909.00000	541909.000000	541909.000000
mean	270954.00000	9.552250	4.611114
std	156435.79785	218.081158	96.759853
min	0.00000	-80995.000000	-11062.060000
25%	135477.00000	1.000000	1.250000
50%	270954.00000	3.000000	2.080000
75%	406431.00000	10.000000	4.130000
max	541908.00000	80995.000000	38970.000000

In [38]:
```
data['CustomerID'].sort_values()
```

Out[38]:
```
437603        0.0
261044        0.0
261045        0.0
261046        0.0
261047        0.0
             ...
198739    18287.0
198738    18287.0
198737    18287.0
198743    18287.0
392725    18287.0
Name: CustomerID, Length: 541909, dtype: object
```

In [41]:
```
'guest_' + data['InvoiceNo'].astype('str')
```

Out[41]:
```
0         guest_536365
1         guest_536365
2         guest_536365
3         guest_536365
4         guest_536365
              ...
541904    guest_581587
541905    guest_581587
541906    guest_581587
541907    guest_581587
541908    guest_581587
Name: InvoiceNo, Length: 541909, dtype: object
```

In [42]:
```
data['InvoiceNo']
```

Out[42]:
```
0         536365
1         536365
2         536365
3         536365
4         536365
           ...
541904    581587
541905    581587
541906    581587
541907    581587
541908    581587
Name: InvoiceNo, Length: 541909, dtype: object
```

In [44]:
```
data.columns
```

Out[44]:
```
Index(['Unnamed: 0', 'InvoiceNo', 'StockCode', 'Quantity', 'InvoiceDate',
       'UnitPrice', 'CustomerID', 'Country'],
      dtype='object')
```

In [46]:
```
data['Quantity'].describe()
```

Out[46]:
```
count    541909.000000
mean          9.552250
std         218.081158
min      -80995.000000
25%           1.000000
50%           3.000000
75%          10.000000
max       80995.000000
Name: Quantity, dtype: float64
```

In [47]:
```
IQR = data.Quantity.describe()['75%'] - data.Quantity.describe()['25%']

low_range = data.Quantity.describe()['25%'] - 1.5*IQR
high_range = data.Quantity.describe()['75%'] + 1.5*IQR

print('Low range  : {} '.format(low_range))
print('High range : {} '.format(high_range))
```

```
Low range  : -12.5
High range : 23.5
```

In [48]:
```
data.Quantity.describe()['75%']
```

Out[48]:
```
10.0
```

In [49]:
```
data.Quantity.describe()['25%']
```

Out[49]:
```
1.0
```

In [50]:
```
data.Quantity.describe()['25%'] - 1.5*IQR
```

Out[50]: -12.5

In [51]:
```
IQR = data.Quantity.describe()['75%'] - data.Quantity.describe()['25%']
```

In [52]:
```
IQR
```

Out[52]: 9.0

In [53]:
```
1.5*IQR
```

Out[53]: 13.5

In [54]:
```
data.Quantity.describe()['75%'] + 1.5*IQR
```

Out[54]: 23.5

In [55]:
```
data = data[(data['Quantity']< 5000) | (data['Quantity']> -5000)]
```

In [56]:
```
data
```

Out[56]:

	Unnamed: 0	InvoiceNo	StockCode	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	0	536365	85123A	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	1	536365	71053	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	2	536365	84406B	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	3	536365	84029G	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	4	536365	84029E	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
...	...	...	...	...	...	...	...	...
541904	541904	581587	22613	12	2011-12-09 12:50:00	0.85	12680.0	France
541905	541905	581587	22899	6	2011-12-09 12:50:00	2.10	12680.0	France
541906	541906	581587	23254	4	2011-12-09 12:50:00	4.15	12680.0	France
541907	541907	581587	23255	4	2011-12-09 12:50:00	4.15	12680.0	France
541908	541908	581587	22138	3	2011-12-09 12:50:00	4.95	12680.0	France

541909 rows × 8 columns

In [57]:
```
data.columns
```

Out[57]:
```
Index(['Unnamed: 0', 'InvoiceNo', 'StockCode', 'Quantity', 'InvoiceDate',
       'UnitPrice', 'CustomerID', 'Country'],
      dtype='object')
```

In [58]:
```
IQR = data.UnitPrice.describe()['75%'] - data.UnitPrice.describe()['25%']
```

In [59]:
```
IQR
```

Out[59]:
```
2.88
```

In [60]:
```
data.UnitPrice.describe()['75%']
```

Out[60]:
```
4.13
```

In [65]:
```
# We will start by first removing the duplicate rows
data.drop_duplicates(inplace=True)

# Dropping rows containing missing values
data.dropna(inplace=True)

# Checking for missing values again
data.isna().sum()
```

Out[65]:
```
Unnamed: 0     0
InvoiceNo      0
StockCode      0
Quantity       0
InvoiceDate    0
UnitPrice      0
CustomerID     0
Country        0
dtype: int64
```

In [66]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 541909 entries, 0 to 541908
Data columns (total 8 columns):
 #   Column       Non-Null Count   Dtype  
---  ------       --------------   -----  
 0   Unnamed: 0   541909 non-null  int64  
 1   InvoiceNo    541909 non-null  object 
 2   StockCode    541909 non-null  object 
 3   Quantity     541909 non-null  int64  
 4   InvoiceDate  541909 non-null  object 
 5   UnitPrice    541909 non-null  float64
 6   CustomerID   541909 non-null  object 
 7   Country      541909 non-null  object 
dtypes: float64(1), int64(2), object(5)
memory usage: 37.2+ MB
```

In [67]:
```
a = np.arange(6)
```

In [68]:
```
a
```

Out[68]:
```
array([0, 1, 2, 3, 4, 5])
```

In [69]:
```
a2 = a[np.newaxis, :]
```

In [70]:
```
a2
```

Out[70]:
```
array([[0, 1, 2, 3, 4, 5]])
```

In [71]:
```
a = np.array([1, 2, 3, 4, 5, 6])
```

In [72]:
```
a
```

Out[72]:
```
array([1, 2, 3, 4, 5, 6])
```

In [73]:
```
a = np.array([[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12]])
```

In [74]:
```
a[0]
```

Out[74]:
```
array([1, 2, 3, 4])
```

In [75]:
```
a = np.array([1, 2, 3])
```

In [76]:
```
a
```

Out[76]:
```
array([1, 2, 3])
```

In [77]:
```
np.zeros(2)
```

Out[77]:
```
array([0., 0.])
```

In [80]:
```
np.ones(2)
```

Out[80]:
```
array([1., 1.])
```

In [82]:
```
np.empty(2)
```

Out[82]:
```
array([1., 1.])
```

In [83]:
```
np.arange(4)
```

Out[83]:
```
array([0, 1, 2, 3])
```

In [84]:
```
np.arange(1,10,2)
```

Out[84]:
```
array([1, 3, 5, 7, 9])
```

In [85]:
```
np.linspace(1,20,5)
```

Out[85]:
```
array([ 1.  ,  5.75, 10.5 , 15.25, 20.  ])
```

In [87]:
```
x=np.ones(4,dtype=np.int64)
```

In [88]:
```
x
```

Out[88]:
```
array([1, 1, 1, 1], dtype=int64)
```

In [89]:
```
arr=np.array([2,3,4,5,7,'tinku'])
```

In [90]:
```
arr
```

Out[90]:
```
array(['2', '3', '4', '5', '7', 'tinku'], dtype='<U11')
```

In [91]:
```
np.sort(arr)
```

Out[91]:
```
array(['2', '3', '4', '5', '7', 'tinku'], dtype='<U11')
```

In [92]:
```
arr2=np.array([2,1,4,3,7,8,6])
```

In [93]:
```
arr2
```

Out[93]:
```
array([2, 1, 4, 3, 7, 8, 6])
```

In [94]:
```
np.sort(arr2)
```

Out[94]:
```
array([1, 2, 3, 4, 6, 7, 8])
```

In [95]:
```
np.argsort(arr2)
```

Out[95]:
```
array([1, 0, 3, 2, 6, 4, 5], dtype=int64)
```

In [97]:
```
arr2[6]
```

Out[97]:
```
6
```

In [98]:
```
np.lexsort(arr2)
```

Out[98]:
```
0
```

In [99]:
```
np.searchsorted(arr2)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_37660/4071567889.py in <module>
----> 1 np.searchsorted(arr2)

<__array_function__ internals> in searchsorted(*args, **kwargs)

TypeError: _searchsorted_dispatcher() missing 1 required positional argument:
 'v'
```

In [101]:
```
a=np.array([1,2,3,4])
b=np.array([4,5,6,7])
np.concatenate((a,b))
```

Out[101]:
```
array([1, 2, 3, 4, 4, 5, 6, 7])
```

In [104]:
```
x=np.array([[1,2,3],[4,5,6]])
y=np.array([[6,7,8],[9,10,12]])
xy=np.concatenate((x,y),axis=0)
```

In [105]:
```
xy
```

Out[105]:
```
array([[ 1,  2,  3],
       [ 4,  5,  6],
       [ 6,  7,  8],
       [ 9, 10, 12]])
```

In [106]:
```
x=np.array([[1,2,3],[4,5,6]])
y=np.array([[6,7,8],[9,10,12]])
xy=np.concatenate((x,y),axis=1)
```

In [107]:
```
xy
```

Out[107]:
```
array([[ 1,  2,  3,  6,  7,  8],
       [ 4,  5,  6,  9, 10, 12]])
```

In [108]:
```
array_example = np.array([[[0, 1, 2, 3],
                           [4, 5, 6, 7]],

                          [[0, 1, 2, 3],
                           [4, 5, 6, 7]],

                          [[0 ,1 ,2, 3],
                           [4, 5, 6, 7]]])
```

In [110]:
```
array_example.ndim
```

Out[110]:
```
3
```

In [111]:
```
array_example.size
```

Out[111]:
```
24
```

In [114]:
```
a=np.arange(6)
```

In [115]:
```
a
```

Out[115]:
```
array([0, 1, 2, 3, 4, 5])
```

In [117]:
```
y=a.reshape(3,2)
```

In [118]:
```
y
```

Out[118]:
```
array([[0, 1],
       [2, 3],
       [4, 5]])
```

In [ ]: