Analysis on ML Test Score

In [1]:

```
import pandas as pd
```

In [2]:

```
data = pd.read_csv('E:/file2/Downloads/scores_data.csv')
data.head()
```

Out[2]:

	Batch	User_ID	Score
0	AI_ELITE_7	uid_149	6 / 7
1	AI_ELITE_7	uid_148	6 / 7
2	AI_ELITE_7	uid_147	7 / 7
3	AI_ELITE_7	uid_146	7 / 7
4	AI_ELITE_7	uid_145	4 / 7

In [3]:

```
data.isnull().sum()
```

Out[3]:

```
Batch          0
User_ID        0
   Score       0
dtype: int64
```

In [10]:

```
data.columns = data.columns.str.replace(" ",'')
```

In [11]:

```
data.isnull().sum()
```

Out[11]:

```
Batch      0
User_ID    0
Score      0
dtype: int64
```

In [12]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 149 entries, 0 to 148
Data columns (total 3 columns):
 #   Column   Non-Null Count  Dtype 
---  ------   --------------  ----- 
 0   Batch    149 non-null    object
 1   User_ID  149 non-null    object
 2   Score    149 non-null    object
dtypes: object(3)
memory usage: 3.6+ KB
```

In [14]:

```
data['Score'].unique()
```

Out[14]:

```
array(['6 / 7', '7 / 7', '4 / 7', '5 / 7', '3 / 7', '2 / 7', '0 / 7',
       '1 / 7'], dtype=object)
```

In [22]:

```
def marks(datas):
    num,deno = datas.split("/")
    return (int(num)/int(deno))*100
```

In [24]:

```
data['Marks']=data['Score'].apply(marks)
```

In [25]:

```
data.head()
```

Out[25]:

	Batch	User_ID	Score	Marks
0	AI_ELITE_7	uid_149	6/7	85.714286
1	AI_ELITE_7	uid_148	6/7	85.714286
2	AI_ELITE_7	uid_147	7/7	100.000000
3	AI_ELITE_7	uid_146	7/7	100.000000
4	AI_ELITE_7	uid_145	4/7	57.142857

In [26]:

```
data['Marks'] = data["Marks"].astype(int)
```

In [27]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 149 entries, 0 to 148
Data columns (total 4 columns):
 #   Column   Non-Null Count  Dtype 
---  ------   --------------  ----- 
 0   Batch    149 non-null    object
 1   User_ID  149 non-null    object
 2   Score    149 non-null    object
 3   Marks    149 non-null    int32 
dtypes: int32(1), object(3)
memory usage: 4.2+ KB
```

In [28]:

```
data.head()
```

Out[28]:

	Batch	User_ID	Score	Marks
0	AI_ELITE_7	uid_149	6/7	85
1	AI_ELITE_7	uid_148	6/7	85
2	AI_ELITE_7	uid_147	7/7	100
3	AI_ELITE_7	uid_146	7/7	100
4	AI_ELITE_7	uid_145	4/7	57

In [29]:

```
print('Mean :',data['Marks'].mean())
print('median :', data['Marks'].median())
```

```
Mean : 62.20134228187919
median : 57.0
```

In [30]:

```
print('minmum:',data["Marks"].min())
print('maxmum:',data["Marks"].max())
```

```
minmum: 0
maxmum: 100
```

In [31]:

```
print('std:',data['Marks'].std())
```

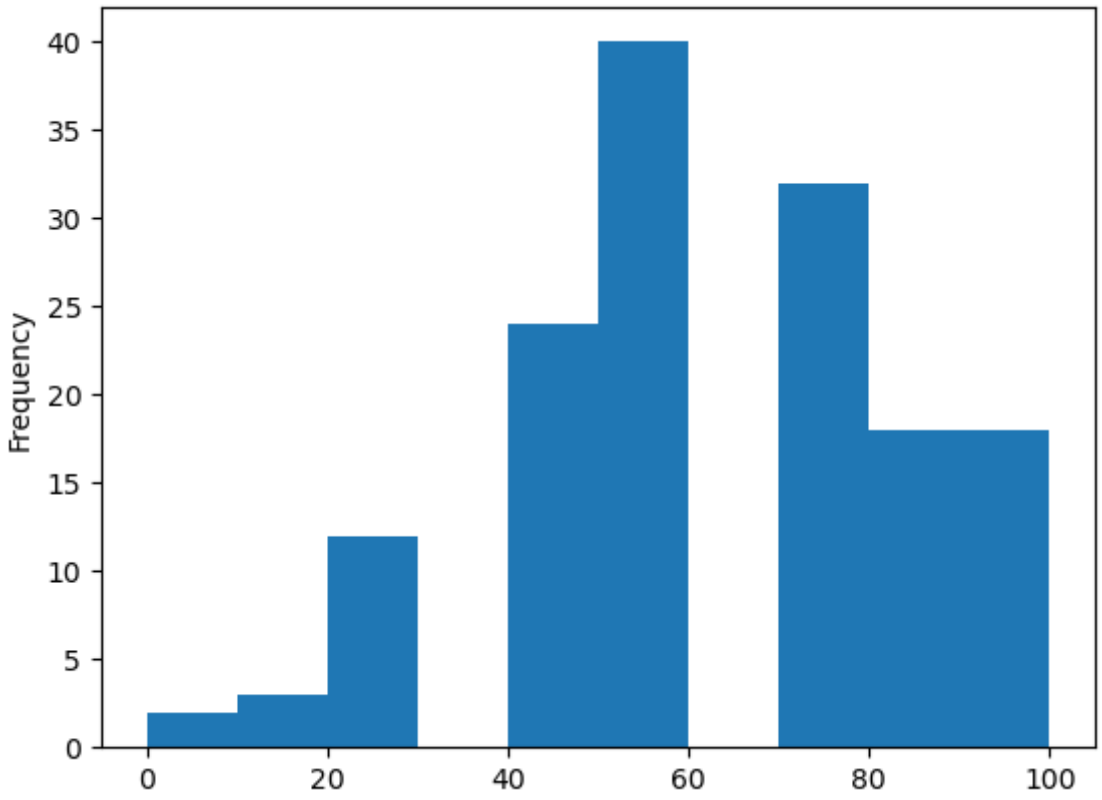
```
std: 22.834889103042936
```

In [37]:

```
data['Marks'].plot(kind='hist')
```

Out[37]:

```
<AxesSubplot:ylabel='Frequency'>
```

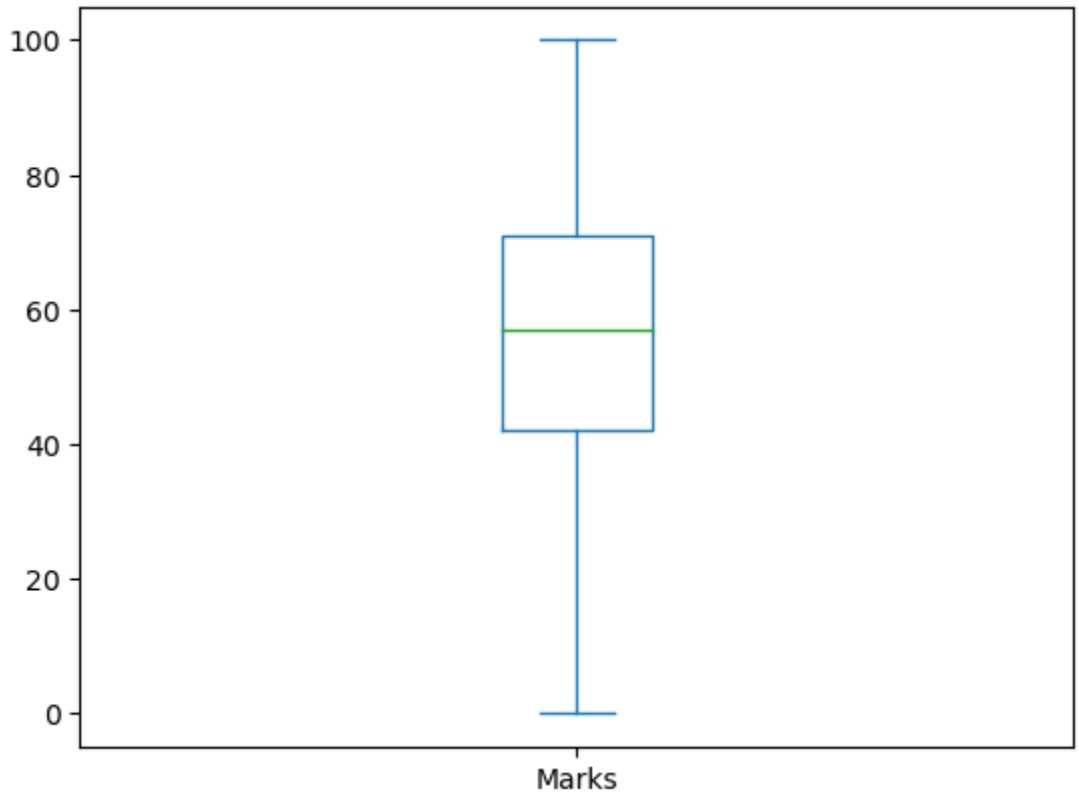


In [38]:

```
data['Marks'].plot(kind='box')
```

Out[38]:

```
<AxesSubplot:>
```



In [51]:

```
import re

def process(pattern):
    batch1 = re.sub('[^0-9]','',pattern)
    return batch1
```

In [52]:

```
data['batches']= data['Batch'].apply(process)
```

In [58]:

```
data['batches']=data['batches'].astype(int)
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 149 entries, 0 to 148
Data columns (total 5 columns):
 #   Column   Non-Null Count  Dtype 
---  ------   --------------  ----- 
 0   Batch    149 non-null    object
 1   User_ID  149 non-null    object
 2   Score    149 non-null    object
 3   Marks    149 non-null    int32 
 4   batches  149 non-null    int32 
dtypes: int32(2), object(3)
memory usage: 4.8+ KB
```

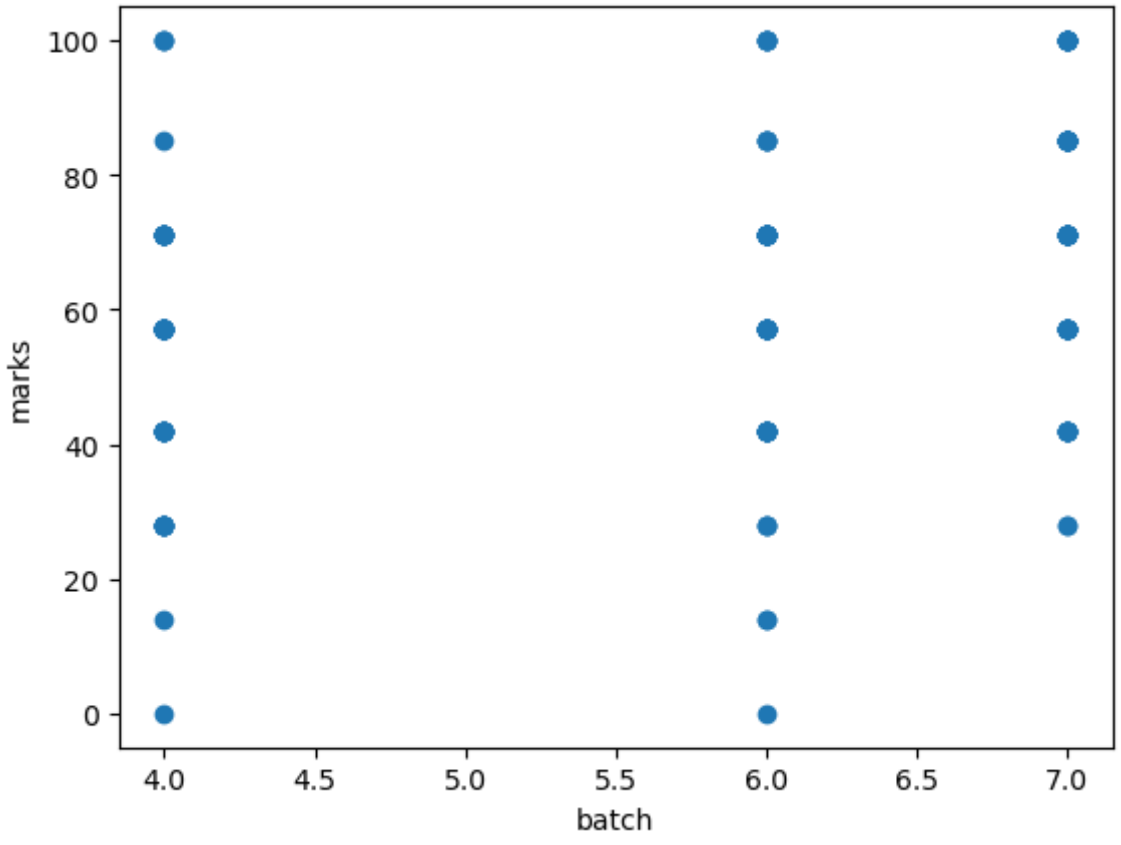
In [66]:

```
import matplotlib.pyplot as pl


plt.scatter(data['batches'],data['Marks'])
plt.xlabel('batch')
plt.ylabel('marks')
```

Out[66]:

```
Text(0, 0.5, 'marks')
```



In [69]:

```
print(data['batches'].value_counts())
print(data['Marks'].value_counts())
```

```
7    53
6    48
4    48
Name: batches, dtype: int64
57     40
71     32
42     24
85     18
100    18
28     12
14      3
0       2
Name: Marks, dtype: int64
```

In [74]:

```
data[['batches','Marks']].value_counts()
```

Out[74]:

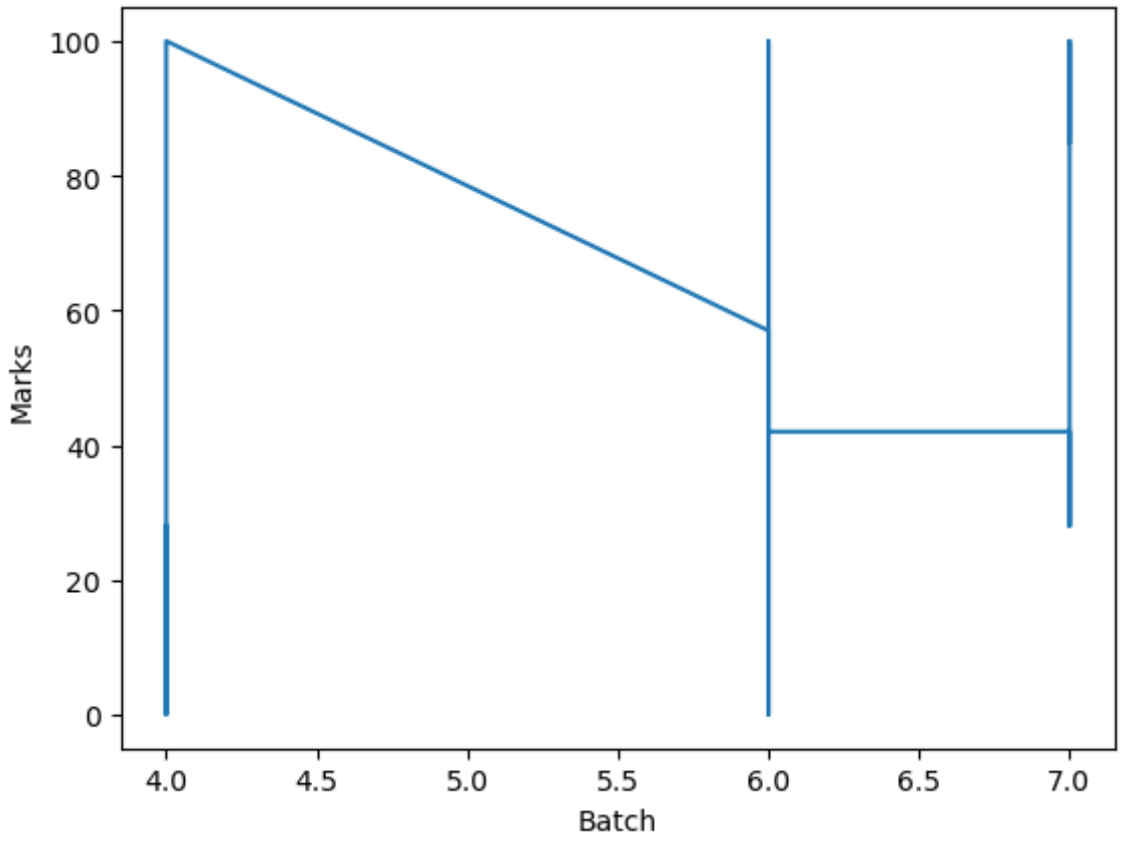
```
batches  Marks
4        57       19
7        85       13
6        71       13
         57       11
7        71       11
         100      10
         57       10
6        42        9
4        71        8
         42        8
7        42        7
4        28        7
6        100       5
         85        4
         28        3
4        100       3
6        14        2
7        28        2
6        0         1
4        14        1
         85        1
         0         1
dtype: int64
```

In [78]:

```
plt.plot(data['batches'],data['Marks'])
plt.xlabel('Batch')
plt.ylabel('Marks')
```

Out[78]:

```
Text(0, 0.5, 'Marks')
```



In [144]:

```
batch1 = data[data['Batch']=='AI_ELITE_7'].sort_values(by = "Marks",ascending=False)
batch2 = data[data['Batch']=='AI_ELITE_6']
batch3 = data[data['Batch']=='AI_ELITE_4']
```

In [135]:

```
batch1.info()
```

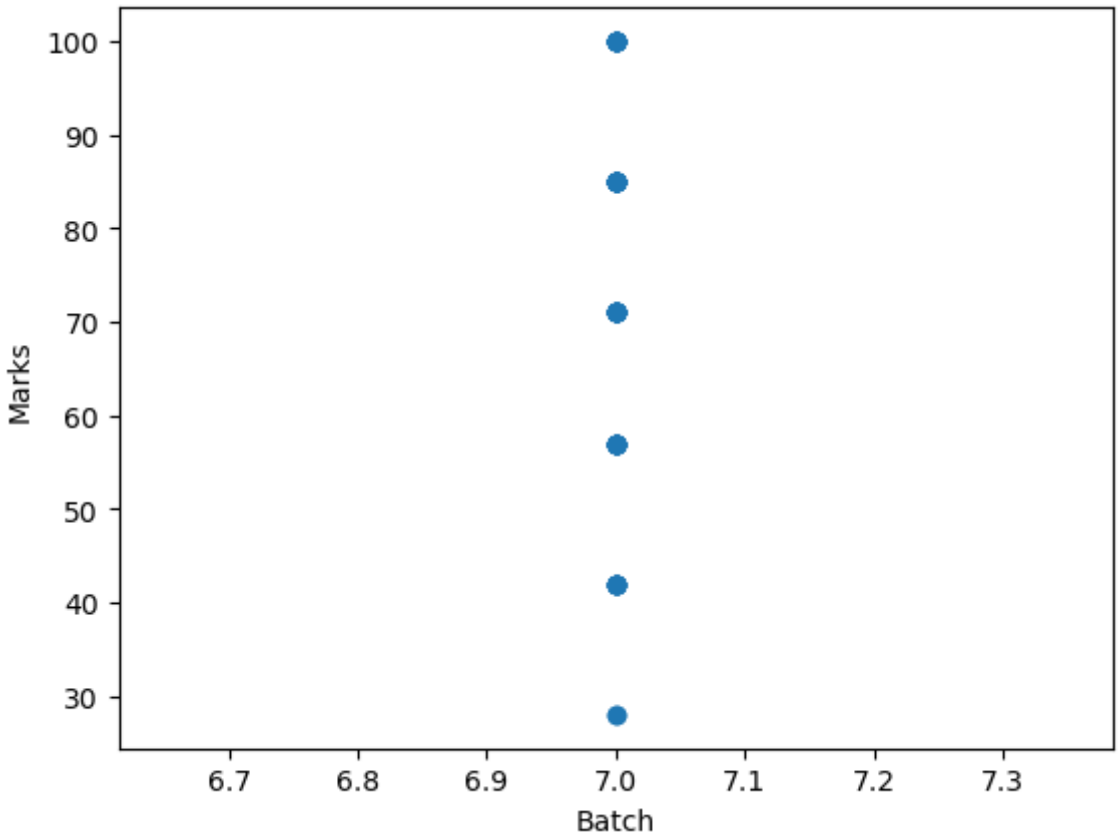
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 53 entries, 50 to 35
Data columns (total 5 columns):
 #   Column   Non-Null Count  Dtype 
---  ------   --------------  ----- 
 0   Batch    53 non-null     object
 1   User_ID  53 non-null     object
 2   Score    53 non-null     object
 3   Marks    53 non-null     int32 
 4   batches  53 non-null     int32 
dtypes: int32(2), object(3)
memory usage: 2.1+ KB
```

In [108]:

```
#batch1['Marks'].plot(kind='bar')
plt.scatter(x=batch1['batches'],y = batch1['Marks'])
plt.xlabel('Batch')
plt.ylabel('Marks')
```

Out[108]:

```
Text(0, 0.5, 'Marks')
```



In [105]:

```
batch2['Marks']
```

Out[105]:

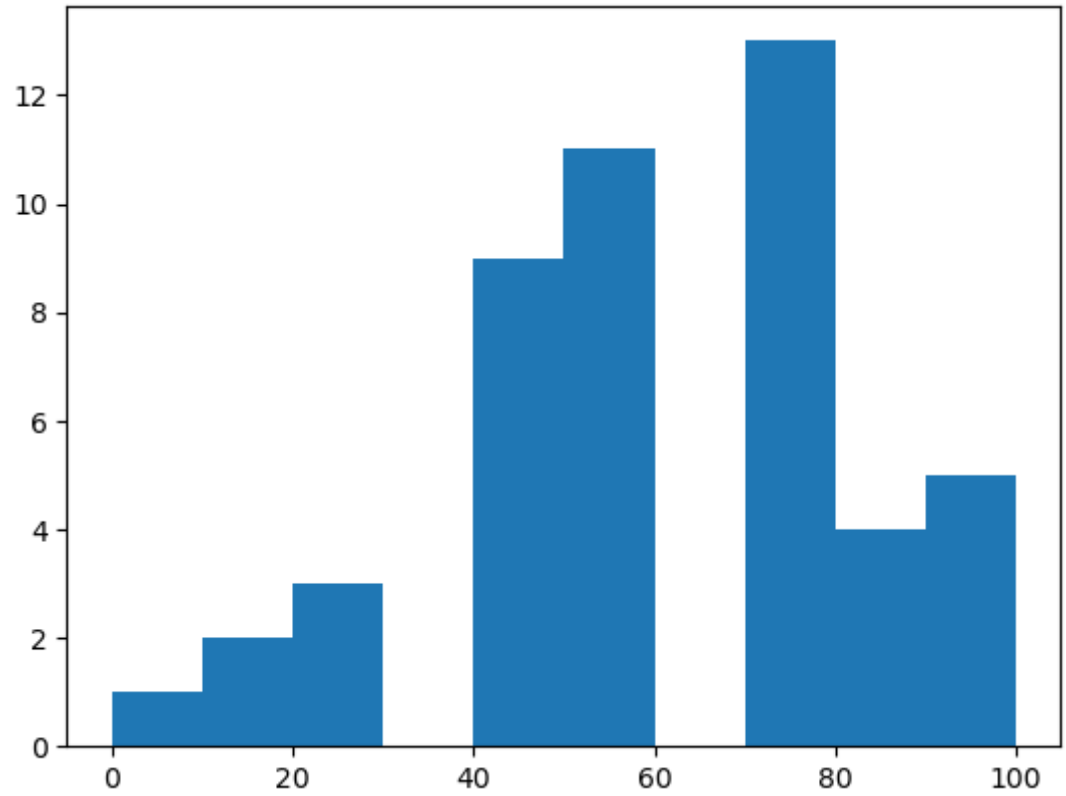
```
53      42
54      57
55      85
56      85
57      57
58      42
59      57
60      42
61      71
62      42
63     100
64      71
65      71
66      71
67     100
68       0
69      71
70      57
71      14
72      57
73      71
74     100
75      57
76      42
77      42
78     100
79      85
80      42
81      85
82      71
83      71
84      57
85      71
86      71
87      71
88      71
89      57
90      42
91      57
92      28
93      28
94      57
95      28
96      42
97      14
98     100
99      71
100     57
Name: Marks, dtype: int32
```

In [109]:

```
plt.hist(batch2['Marks'])
```

Out[109]:

```
(array([ 1.,  2.,  3.,  0.,  9., 11.,  0., 13.,  4.,  5.]),
 array([  0.,  10.,  20.,  30.,  40.,  50.,  60.,  70.,  80.,  90., 10
0.]),
 <BarContainer object of 10 artists>)
```



In [115]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 149 entries, 0 to 148
Data columns (total 5 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   Batch    149 non-null    object
 1   User_ID  149 non-null    object
 2   Score    149 non-null    object
 3   Marks    149 non-null    int32
 4   batches  149 non-null    int32
dtypes: int32(2), object(3)
memory usage: 4.8+ KB
```

In [119]:

```
data['batches'].value_counts()
```

Out[119]:

```
7    53
6    48
4    48
Name: batches, dtype: int64
```

In [137]:

```
batch1['Marks'].value_counts()
```

Out[137]:

```
85     13
71     11
57     10
100    10
42      7
28      2
Name: Marks, dtype: int64
```

In [124]:

```
batch2["Marks"].value_counts()
```

Out[124]:

```
71     13
57     11
42      9
100     5
85      4
28      3
14      2
0       1
Name: Marks, dtype: int64
```

In [125]:

```
batch3["Marks"].value_counts()
```

Out[125]:

```
57     19
71      8
42      8
28      7
100     3
85      1
14      1
0       1
Name: Marks, dtype: int64
```

In [179]:

```
## printing highest marks in each batch
top_mark = []
top_mark.append(batch1[batch1['Marks'] == batch1['Marks'].max()].count()[0])
top_mark.append(batch2[batch2['Marks'] == batch1['Marks'].max()].count()[0])
top_mark.append(batch3[batch3['Marks'] == batch1['Marks'].max()].count()[0])
print(top_mark)
```

...

In [184]:

```
labels = [f"Batch{i+1} - {top_mark[i]} students" for i in range(len(top_mark))]
colors = ['red','orange','green']
explode = [0.2,0,0]
plt.pie( top_mark, labels = labels, colors = colors, shadow = True,autopct = '%1.2f%%',ex
plt.show()
```

...

In [200]:

```
## Batches with highest pass ratio

## Batch 1
batch_1_pass_count = batch1[batch1['Marks'] >= 40].count()[0]
batch_1_total_count = batch1['Marks'].shape[0]
batch_1_pass_ratio = (batch_1_pass_count/ batch_1_total_count) * 100
print(batch_1_pass_ratio)

## Batch 2
batch_2_pass_count = batch2[batch2['Marks'] >= 40].count()[0]
batch_2_total_count = batch2['Marks'].shape[0]
batch_2_pass_ratio = (batch_2_pass_count/ batch_2_total_count) * 100
print(batch_2_pass_ratio)

## Batch 3
batch_3_pass_count = batch3[batch3['Marks'] >= 40].count()[0]
batch_3_total_count = batch3['Marks'].shape[0]
batch_3_pass_ratio = (batch_3_pass_count/ batch_3_total_count) * 100
print(batch_1_pass_ratio)
```

```
96.22641509433963
87.5
96.22641509433963
```

In [201]:

```
pass_ratio = [batch_1_pass_ratio, batch_2_pass_ratio,batch_1_pass_ratio]
```

In [204]:

```
pass_ratio
```

Out[204]:

```
[96.22641509433963, 87.5, 96.22641509433963]
```

In [208]:

```
print('AI_ELITE_7 Score :',batch_1_pass_ratio)
print('AI_ELITE_6 Score :',batch_2_pass_ratio)
print('AI_ELITE_4 Score :',batch_3_pass_ratio)
```

```
AI_ELITE_7 Score : 96.22641509433963
AI_ELITE_6 Score : 87.5
AI_ELITE_4 Score : 81.25
```

In [ ]: