In [14]:

```python
import pandas as pd
df = pd.read_csv('iris.csv')
df.head()
```

Out[14]:

	SepalLength	SepalWidth	PetalLength	PetalWidth	Name
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa

In [14]:

```python
import pandas as pd
df = pd.read_csv('iris.csv')
df.head()
```

Out[14]:

	SepalLength	SepalWidth	PetalLength	PetalWidth	Name
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa

In [14]:

```python
import pandas as pd
df = pd.read_csv('iris.csv')
df.head()
```

Out[14]:

	SepalLength	SepalWidth	PetalLength	PetalWidth	Name
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa

In [12]:

```python
import numpy as np
df.size
```

Out[12]:

```
750
```

There are 4 features and 1 class. The features are numeric.

In [19]:

```python
# SUMMARY:
```

```
# SUMMARY:
df.describe()
```

Out[19]:

	SepalLength	SepalWidth	PetalLength	PetalWidth
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.054000	3.758667	1.198667
std	0.828066	0.433594	1.764420	0.763161
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

In [21]:

```
# VARIANCE:
df[df.columns[0:4]].var()
```

Out[21]:

```
SepalLength    0.685694
SepalWidth     0.188004
PetalLength    3.113179
PetalWidth     0.582414
dtype: float64
```

In [25]:

```
# ABSOLUTE MAXIMUM DIFFERENCE
df[df.columns[0:4]].max()-df[df.columns[0:4]].dropna().min()
```

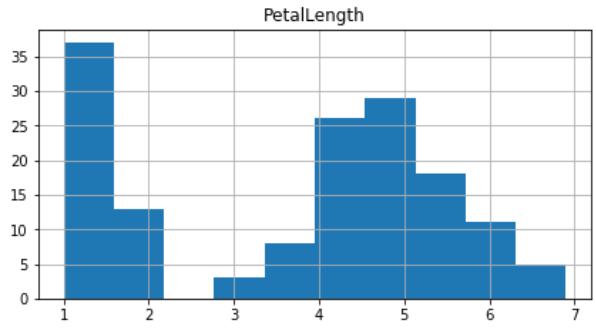
Out[25]:

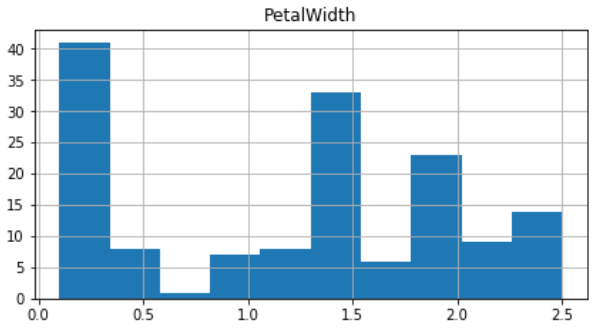
```
SepalLength    3.6
SepalWidth     2.4
PetalLength    5.9
PetalWidth     2.4
dtype: float64
```

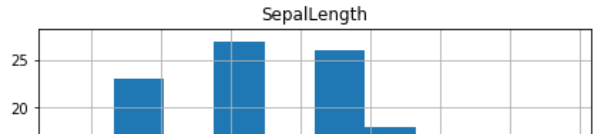
1.2 DATA VISUALISATION

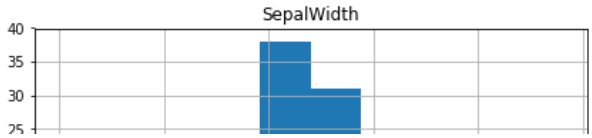
In [33]:

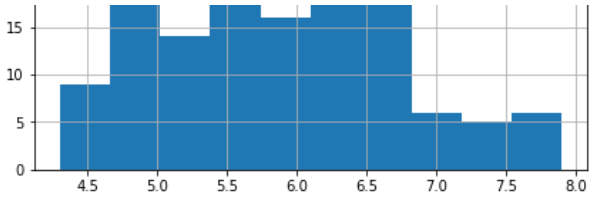
```
# HISTOGRAM
hist1 = df.iloc[:,0:4].hist(figsize=(16,8))
```

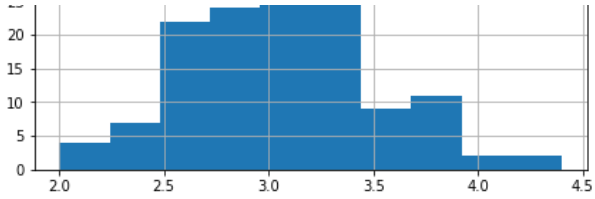






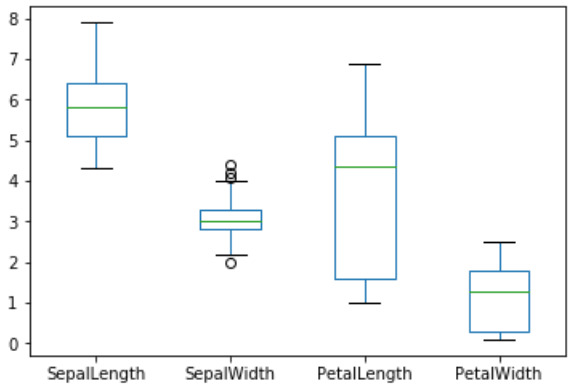






In [35]:

```
# BOXPLOT
bp = df.boxplot(grid=False)
```



Pen-based Handwritten Digits Dataset

In [52]:

```
import pandas as pd
Digits = pd.read_csv('pendigits.csv', header = None)
```

In [53]:

```
Digits.head()
```

Out[53]:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	47	100	27	81	57	37	26	0	0	23	56	53	100	90	40	98	8
1	0	89	27	100	42	75	29	45	15	15	37	0	69	2	100	6	2
2	0	57	31	68	72	90	100	100	76	75	50	51	28	25	16	0	1
3	0	100	7	92	5	68	19	45	86	34	100	45	74	23	67	0	4
4	0	67	49	83	100	100	81	80	60	60	40	40	33	20	47	0	1

There are 16 features and 1 class Features are numeric while class is nominal

In [54]:

```
Digits.describe()
```

Out[54]:

	0	1	2	3	4	5	6	7	8	
count	7494.000000	7494.000000	7494.000000	7494.000000	7494.000000	7494.000000	7494.000000	7494.000000	7494.000000	7494.00
mean	37.384307	84.679343	40.005604	82.889512	50.878303	65.044436	51.471844	44.599680	57.129971	34.06
std	33.322024	16.848420	26.256025	19.638582	34.927201	27.377341	30.680075	30.659478	33.680340	27.45
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00
25%	5.000000	76.000000	20.000000	70.000000	17.000000	48.000000	28.000000	22.000000	30.000000	7.00
50%	31.000000	89.000000	39.000000	89.000000	56.000000	71.000000	54.000000	42.000000	60.000000	33.00

	0	1	2	3	4	5	6	7	8	
75%	61.000000	100.000000	58.000000	100.000000	81.000000	86.000000	75.000000	65.000000	88.000000	55.00
max	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.00

In [41]:

```
# Absolute maximum difference

Digits[Digits.columns].max()-Digits[Digits.columns].min()
```

Out[41]:

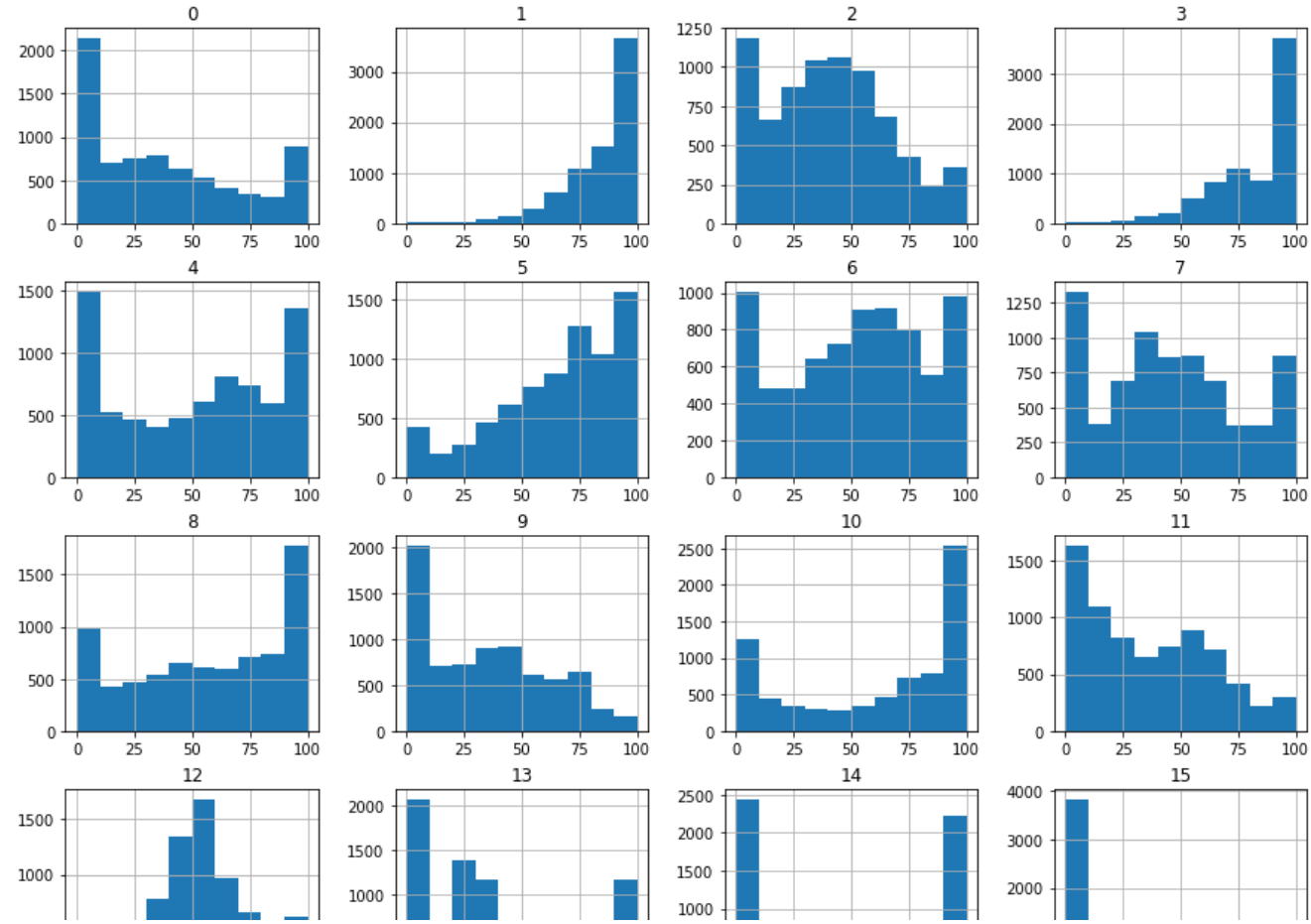
```
0      100
1      100
2      100
3      100
4      100
5      100
6      100
7      100
8      100
9      100
10     100
11     100
12     100
13     100
14     100
15     100
16       9
dtype: int64
```

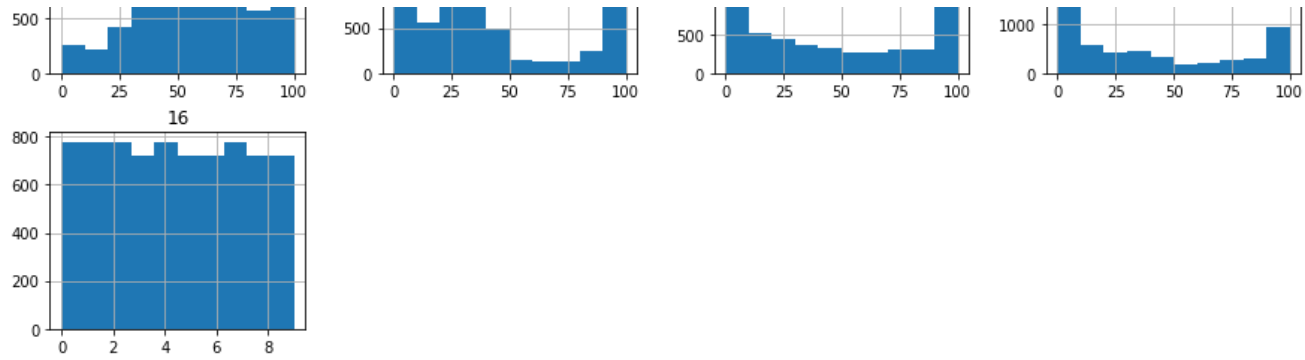
DATA VISUALISATION

In [59]:

```
# HISTOGRAM

hist2 = Digits.hist(figsize=(15,15))
```

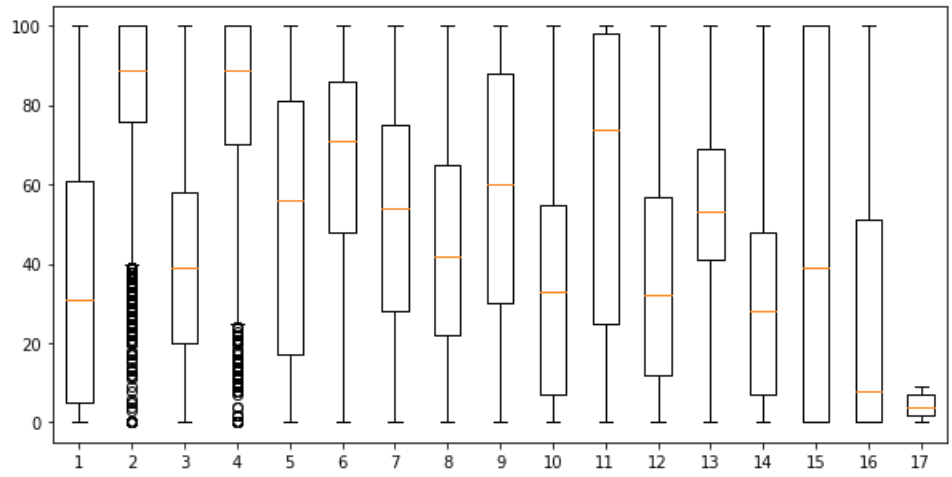




In [58]:

```
# BOXPLOT

import matplotlib.pyplot as plt
fig = plt.figure(figsize=(10,5))
ax = fig.add_subplot(111)
npbox = ax.boxplot(Digits.values)
```



In [ ]:

```

```

A1 - The histograms for petal length and petal width contains points of high frequency, while the histogram for sepal length and sepal width has distributions that has highest frequency at a particular point. More importantly, the petal length and petal width histograms display bimodal distributions, whereas the sepal length and sepal width histograms display normal distributions.

A2 - Given the petal width histogram, a value between 2.5-3.0 would be effective at segmented the distribution of petal widths, as the histogram displays a low frequency within this range.

A3 - Based upon the boxplots generated for the Iris dataset, sepal length and petal width have significantly different medians, as there is no overlap between the ranges of these two boxplots. Strictly speaking, several other pairs of varibles can be considered to have signficantly different medians; sepal length and petal width simply have the greatest difference.

A4 -Based solely upon the box plots, petal length appears to explain the greatest amount of the data, as it displays the greatest range -- the greatest variability -- among the features.

B - (1,2,3) Given the boxplots generated for the Digits dataset, we do observe outliers particularly for features labeled 2 and 4, respectively. The histograms for these features display skewed distributions, with lower frequency of smaller values. Given these observations, the outliers for both features can be explained by mean values skewed to larger values, resulting in the smaller values in the distribution being considered outliers by the box plots.