

In [204]:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
import math
from scipy.stats import normaltest
from scipy.stats import ttest_1samp
from scipy import stats
from scipy.stats import norm
from statsmodels.stats import weightstats as stests
from statsmodels.stats.proportion import proportions_ztest

df = pd.read_csv("Gandum.csv", names=["id", "Daerah", "SumbuUtama", "SumbuKecil",
    "Keunikan", "AreaBulatan", "Diameter", "KadarAir", "Keliling", "Bulatan", "Ran
sum", "Kelas"])
```

In [205]:

```
print("SOAL 1\n")

print("----- DAERAH -----")
print("Rata-rata:",df["Daerah"].mean())
print("Median:",df["Daerah"].median())
print("Modus:")
print(df["Daerah"].mode())
print("Standar deviasi:",df["Daerah"].std())
print("Variansi:",df["Daerah"].var())
range = df["Daerah"].max()-df["Daerah"].min()
print("Range:",range)
print("Nilai minimum:",df["Daerah"].min())
print("Nilai maksimum:",df["Daerah"].max())
print("Kuartil:")
print(df["Daerah"].quantile([0,0.25,0.50,0.75,1]))
Q1 = df["Daerah"].quantile(0.25)
Q3 = df["Daerah"].quantile(0.75)
IQR = Q3-Q1
print("IQR:",IQR)
print("Skewness:",df["Daerah"].skew())
print("Kurtosis:",df["Daerah"].kurt())
print()

print("----- SUMBU UTAMA -----")
print("Rata-rata:",df["SumbuUtama"].mean())
print("Median:",df["SumbuUtama"].median())
print("Modus:", "Data unik sehingga tidak ada modus")
print("Standar deviasi:",df["SumbuUtama"].std())
print("Variansi:",df["SumbuUtama"].var())
range = df["SumbuUtama"].max()-df["SumbuUtama"].min()
print("Range:",range)
print("Nilai minimum:",df["SumbuUtama"].min())
print("Nilai maksimum:",df["SumbuUtama"].max())
print("Kuartil:")
print(df["SumbuUtama"].quantile([0,0.25,0.50,0.75,1]))
Q1 = df["SumbuUtama"].quantile(0.25)
Q3 = df["SumbuUtama"].quantile(0.75)
IQR = Q3-Q1
print("IQR:",IQR)
print("Skewness:",df["SumbuUtama"].skew())
print("Kurtosis:",df["SumbuUtama"].kurt())
print()

print("----- SUMBU KECIL -----")
print("Rata-rata:",df["SumbuKecil"].mean())
print("Median:",df["SumbuKecil"].median())
print("Modus:", "Data unik sehingga tidak ada modus")
print("Standar deviasi:",df["SumbuKecil"].std())
print("Variansi:",df["SumbuKecil"].var())
range = df["SumbuKecil"].max()-df["SumbuKecil"].min()
print("Range:",range)
print("Nilai minimum:",df["SumbuKecil"].min())
print("Nilai maksimum:",df["SumbuKecil"].max())
print("Kuartil:")
print(df["SumbuKecil"].quantile([0,0.25,0.50,0.75,1]))
Q1 = df["SumbuKecil"].quantile(0.25)
Q3 = df["SumbuKecil"].quantile(0.75)
IQR = Q3-Q1
print("IQR:",IQR)
```

```
print("Skewness:",df["SumbuKecil"].skew())
print("Kurtosis:",df["SumbuKecil"].kurt())
print()

print("----- KEUNIKAN -----")
print("Rata-rata:",df["Keunikan"].mean())
print("Median:",df["Keunikan"].median())
print("Modus:", "Data unik sehingga tidak ada modus")
print("Standar deviasi:",df["Keunikan"].std())
print("Variansi:",df["Keunikan"].var())
range = df["Keunikan"].max()-df["Keunikan"].min()
print("Range:",range)
print("Nilai minimum:",df["Keunikan"].min())
print("Nilai maksimum:",df["Keunikan"].max())
print("Kuartil:")
print(df["Keunikan"].quantile([0,0.25,0.50,0.75,1]))
Q1 = df["Keunikan"].quantile(0.25)
Q3 = df["Keunikan"].quantile(0.75)
IQR = Q3-Q1
print("IQR:",IQR)
print("Skewness:",df["Keunikan"].skew())
print("Kurtosis:",df["Keunikan"].kurt())
print()

print("----- AREA BULATAN -----")
print("Rata-rata:",df["AreaBulatan"].mean())
print("Median:",df["AreaBulatan"].median())
print("Modus:")
print(df["AreaBulatan"].mode())
print("Standar deviasi:",df["AreaBulatan"].std())
print("Variansi:",df["AreaBulatan"].var())
range = df["AreaBulatan"].max()-df["AreaBulatan"].min()
print("Range:",range)
print("Nilai minimum:",df["AreaBulatan"].min())
print("Nilai maksimum:",df["AreaBulatan"].max())
print("Kuartil:")
print(df["AreaBulatan"].quantile([0,0.25,0.50,0.75,1]))
Q1 = df["AreaBulatan"].quantile(0.25)
Q3 = df["AreaBulatan"].quantile(0.75)
IQR = Q3-Q1
print("IQR:",IQR)
print("Skewness:",df["AreaBulatan"].skew())
print("Kurtosis:",df["AreaBulatan"].kurt())
print()

print("----- DIAMETER -----")
print("Rata-rata:",df["Diameter"].mean())
print("Median:",df["Diameter"].median())
print("Modus:")
print(df["Diameter"].mode())
print("Standar deviasi:",df["Diameter"].std())
print("Variansi:",df["Diameter"].var())
range = df["Diameter"].max()-df["Diameter"].min()
print("Range:",range)
print("Nilai minimum:",df["Diameter"].min())
print("Nilai maksimum:",df["Diameter"].max())
print("Kuartil:")
print(df["Diameter"].quantile([0,0.25,0.50,0.75,1]))
Q1 = df["Diameter"].quantile(0.25)
Q3 = df["Diameter"].quantile(0.75)
IQR = Q3-Q1
```

```

print("IQR:", IQR)
print("Skewness:", df["Diameter"].skew())
print("Kurtosis:", df["Diameter"].kurt())
print()

print("----- KADAR AIR -----")
print("Rata-rata:", df["KadarAir"].mean())
print("Median:", df["KadarAir"].median())
print("Modus:")
print(df["KadarAir"].mode())
print("Standar deviasi:", df["KadarAir"].std())
print("Variansi:", df["KadarAir"].var())
range = df["KadarAir"].max() - df["KadarAir"].min()
print("Range:", range)
print("Nilai minimum:", df["KadarAir"].min())
print("Nilai maksimum:", df["KadarAir"].max())
print("Kuartil:")
print(df["KadarAir"].quantile([0, 0.25, 0.50, 0.75, 1]))
Q1 = df["KadarAir"].quantile(0.25)
Q3 = df["KadarAir"].quantile(0.75)
IQR = Q3 - Q1
print("IQR:", IQR)
print("Skewness:", df["KadarAir"].skew())
print("Kurtosis:", df["KadarAir"].kurt())
print()

print("----- KELILING -----")
print("Rata-rata:", df["Keliling"].mean())
print("Median:", df["Keliling"].median())
print("Modus:", "Data unik sehingga tidak ada modus")
print("Standar deviasi:", df["Keliling"].std())
print("Variansi:", df["Keliling"].var())
range = df["Keliling"].max() - df["Keliling"].min()
print("Range:", range)
print("Nilai minimum:", df["Keliling"].min())
print("Nilai maksimum:", df["Keliling"].max())
print("Kuartil:")
print(df["Keliling"].quantile([0, 0.25, 0.50, 0.75, 1]))
Q1 = df["Keliling"].quantile(0.25)
Q3 = df["Keliling"].quantile(0.75)
IQR = Q3 - Q1
print("IQR:", IQR)
print("Skewness:", df["Keliling"].skew())
print("Kurtosis:", df["Keliling"].kurt())
print()

print("----- BULATAN -----")
print("Rata-rata:", df["Bulatan"].mean())
print("Median:", df["Bulatan"].median())
print("Modus:", "Data unik sehingga tidak ada modus")
print("Standar deviasi:", df["Bulatan"].std())
print("Variansi:", df["Bulatan"].var())
range = df["Bulatan"].max() - df["Bulatan"].min()
print("Range:", range)
print("Nilai minimum:", df["Bulatan"].min())
print("Nilai maksimum:", df["Bulatan"].max())
print("Kuartil:")
print(df["Bulatan"].quantile([0, 0.25, 0.50, 0.75, 1]))
Q1 = df["Bulatan"].quantile(0.25)
Q3 = df["Bulatan"].quantile(0.75)
IQR = Q3 - Q1

```

```
print("IQR:", IQR)
print("Skewness:", df["Bulatan"].skew())
print("Kurtosis:", df["Bulatan"].kurt())
print()

print("----- RANSUM -----")
print("Rata-rata:", df["Ransum"].mean())
print("Median:", df["Ransum"].median())
print("Modus:", "Data unik sehingga tidak ada modus")
print("Standar deviasi:", df["Ransum"].std())
print("Variansi:", df["Ransum"].var())
range = df["Ransum"].max() - df["Ransum"].min()
print("Range:", range)
print("Nilai minimum:", df["Ransum"].min())
print("Nilai maksimum:", df["Ransum"].max())
print("Kuartil:")
print(df["Ransum"].quantile([0, 0.25, 0.50, 0.75, 1]))
Q1 = df["Ransum"].quantile(0.25)
Q3 = df["Ransum"].quantile(0.75)
IQR = Q3 - Q1
print("IQR:", IQR)
print("Skewness:", df["Ransum"].skew())
print("Kurtosis:", df["Ransum"].kurt())
```

SOAL 1

----- DAERAH -----

Rata-rata: 4801.246

Median: 4735.0

Modus:

0 3992

1 4881

2 5642

3 6083

dtype: int64

Standar deviasi: 986.3954914816018

Variansi: 972976.0656152307

Range: 4931

Nilai minimum: 2522

Nilai maksimum: 7453

Kuartil:

0.00 2522.00

0.25 4042.75

0.50 4735.00

0.75 5495.50

1.00 7453.00

Name: Daerah, dtype: float64

IQR: 1452.75

Skewness: 0.23814408738280812

Kurtosis: -0.4346305340273977

----- SUMBU UTAMA -----

Rata-rata: 116.04517136778003

Median: 115.40513994999999

Modus: Data unik sehingga tidak ada modus

Standar deviasi: 18.28262595755936

Variansi: 334.25441190402336

Range: 153.79546865999998

Nilai minimum: 74.13311404

Nilai maksimum: 227.9285827

Kuartil:

0.00 74.133114

0.25 104.116098

0.50 115.405140

0.75 129.046792

1.00 227.928583

Name: SumbuUtama, dtype: float64

IQR: 24.93069385000001

Skewness: 0.7615287378076631

Kurtosis: 4.330533548436968

----- SUMBU KECIL -----

Rata-rata: 53.71524598896002

Median: 53.731198774999996

Modus: Data unik sehingga tidak ada modus

Standar deviasi: 4.071074752475035

Variansi: 16.573649640239662

Range: 29.071182429999993

Nilai minimum: 39.90651744

Nilai maksimum: 68.97769987

Kuartil:

0.00 39.906517

0.25 51.193576

0.50 53.731199

0.75 56.325158

1.00 68.977700
Name: SumbuKecil, dtype: float64
IQR: 5.131581650000001
Skewness: -0.010828051555611359
Kurtosis: 0.47556845043813656

----- KEUNIKAN -----

Rata-rata: 0.878763914340001
Median: 0.8900454185
Modus: Data unik sehingga tidak ada modus
Standar deviasi: 0.03658590550477702
Variansi: 0.0013385284816044736
Range: 0.19408517999999997
Nilai minimum: 0.7199162259999999
Nilai maksimum: 0.9140014059999999
Kuartil:
0.00 0.719916
0.25 0.863676
0.50 0.890045
0.75 0.907578
1.00 0.914001
Name: Keunikan, dtype: float64
IQR: 0.04390216424999993
Skewness: -1.6234718222806501
Kurtosis: 2.917255925694389

----- AREA BULATAN -----

Rata-rata: 4937.048
Median: 4857.0
Modus:
0 3802
1 4913
dtype: int64
Standar deviasi: 1011.6962549701573
Variansi: 1023529.3123206415
Range: 5141
Nilai minimum: 2579
Nilai maksimum: 7720
Kuartil:
0.00 2579.00
0.25 4170.25
0.50 4857.00
0.75 5654.25
1.00 7720.00
Name: AreaBulatan, dtype: float64
IQR: 1484.0
Skewness: 0.2575600053152032
Kurtosis: -0.40968492037366033

----- DIAMETER -----

Rata-rata: 77.77115780832005
Median: 77.64527658
Modus:
0 71.293564
1 78.833256
2 84.756224
3 88.006342
dtype: float64
Standar deviasi: 8.056867291849715
Variansi: 64.91311055847777
Range: 40.747172240000005

Nilai minimum: 56.66665803
Nilai maksimum: 97.41383027
Kuartil:
0.00 56.666658
0.25 71.745308
0.50 77.645277
0.75 83.648598
1.00 97.413830
Name: Diameter, dtype: float64
IQR: 11.903290020000014
Skewness: 0.002724966865193717
Kurtosis: -0.4664545130312141

----- KADAR AIR -----
Rata-rata: 0.6483716718980002
Median: 0.626116699
Modus:
0 0.735849
1 0.824405
dtype: float64
Standar deviasi: 0.0943670980937953
Variansi: 0.008905149202643984
Range: 0.46897193099999995
Nilai minimum: 0.409927152
Nilai maksimum: 0.878899083
Kuartil:
0.00 0.409927
0.25 0.572632
0.50 0.626117
0.75 0.726633
1.00 0.878899
Name: KadarAir, dtype: float64
IQR: 0.15400088725000005
Skewness: 0.49366131797330265
Kurtosis: -0.7403261705867821

----- KELILING -----
Rata-rata: 281.479722
Median: 280.04549999999995
Modus: Data unik sehingga tidak ada modus
Standar deviasi: 37.335401715143995
Variansi: 1393.9322212311772
Range: 291.822
Nilai minimum: 197.015
Nilai maksimum: 488.837
Kuartil:
0.00 197.0150
0.25 255.8830
0.50 280.0455
0.75 306.0625
1.00 488.8370
Name: Keliling, dtype: float64
IQR: 50.179500000000002
Skewness: 0.7336269072005543
Kurtosis: 2.272684731245571

----- BULATAN -----
Rata-rata: 0.7617374363080005
Median: 0.7612884395
Modus: Data unik sehingga tidak ada modus
Standar deviasi: 0.0617024607867326


```

Variansi: 0.0038071936671382743
Range: 0.7301581349999999
Nilai minimum: 0.174590178
Nilai maksimum: 0.9047483129999999
Kuartil:
0.00    0.174590
0.25    0.731991
0.50    0.761288
0.75    0.796361
1.00    0.904748
Name: Bulatan, dtype: float64
IQR: 0.06437024175000006
Skewness: -3.5992367663616407
Kurtosis: 29.97509590463005

```

```

----- RANSUM -----
Rata-rata: 2.150915331084002
Median: 2.1935990365
Modus: Data unik sehingga tidak ada modus
Standar deviasi: 0.24976688927065827
Variansi: 0.06238349897594127
Range: 1.0240129659999997
Nilai minimum: 1.440795615
Nilai maksimum: 2.4648085809999998
Kuartil:
0.00    1.440796
0.25    1.983939
0.50    2.193599
0.75    2.381612
1.00    2.464809
Name: Ransum, dtype: float64
IQR: 0.3976734275
Skewness: -0.6581880925333653
Kurtosis: -0.4286557930626156

```

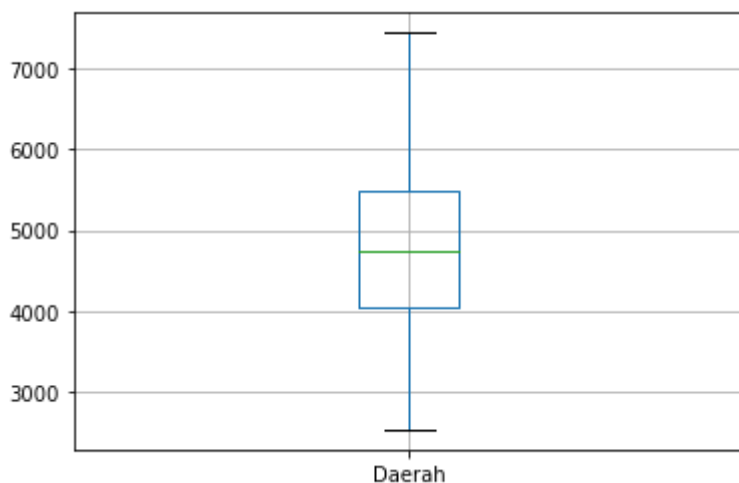
In [206]:

```

print("SOAL 2: Boxplot Daerah")
%matplotlib inline
boxplot = df.boxplot(column=["Daerah"])
# df["Daerah"].plot.hist(bins=100,figsize=(12,8),rwidth=0.5)

```

SOAL 2: Boxplot Daerah



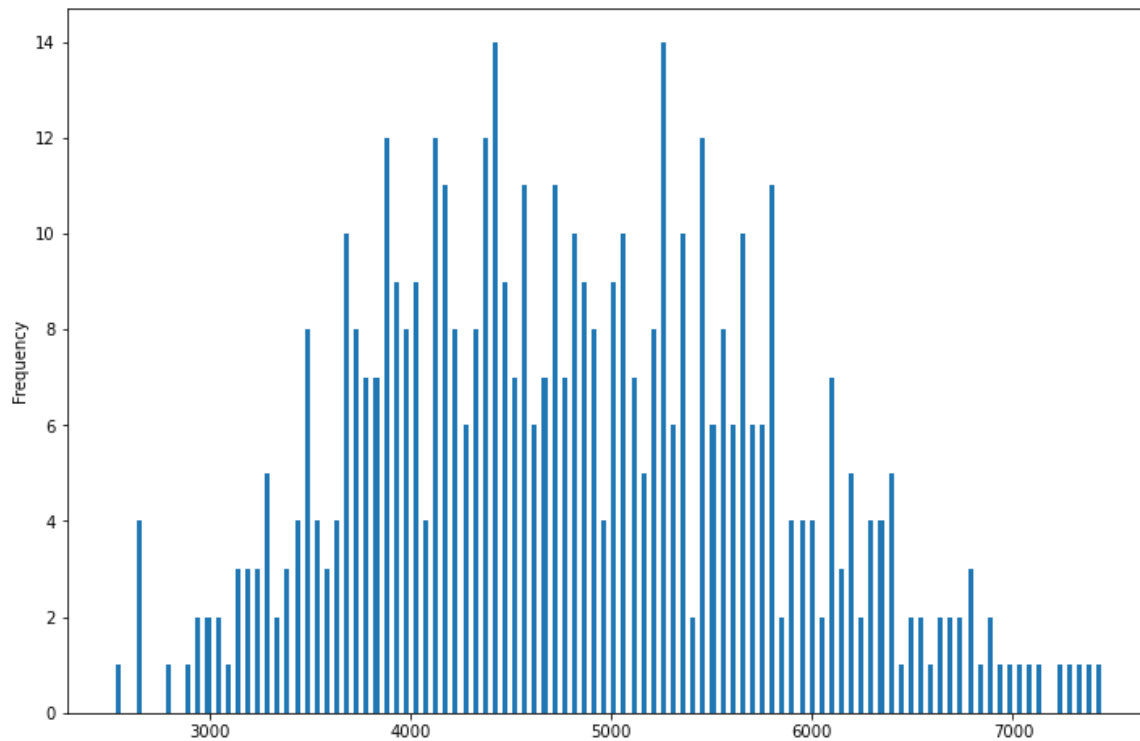
In [207]:

```
print("SOAL 2: Histogram Daerah")  
df["Daerah"].plot.hist(bins=100,figsize=(12,8),rwidth=0.5)
```

SOAL 2: Histogram Daerah

Out[207]:

<matplotlib.axes._subplots.AxesSubplot at 0x7fd511c5cfd0>



In [208]:

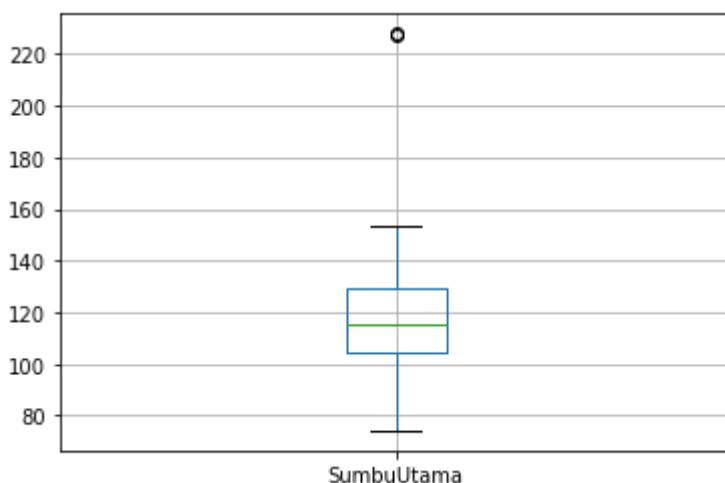
```
print("Kolom Daerah memiliki nilai minimum 2522, Q1 4042.75, median 4735.00, Q3 5495.50, dan nilai maksimum 7453.")
print("Hal ini divisualisasikan pada garis-garis di BoxPlot.")
print("Jarak antarkuartil, IQR = Q3 - Q1, bernilai 1452.75, yaitu bagian kotak pada BoxPlot.")
print("Kolom ini memiliki rata-rata 4801.246.")
print("Kolom ini memiliki standar deviasi 986.3954914816017, variansi 972976.0656152305, dan range 4931.")
print("Kolom ini memiliki nilai skewness 0.23814408738280812, kurva positive skew (kurva condong ke kiri, sebagian besar nilai berada di nilai rendah).")
print("Kolom ini memiliki nilai kurtosis -0.4346305340273977, kurva platykurtic (nilai <3).")
print("Dari BoxPlot, dapat dilihat bahwa tidak ada outlier pada kolom Daerah.")
```

Kolom Daerah memiliki nilai minimum 2522, Q1 4042.75, median 4735.00, Q3 5495.50, dan nilai maksimum 7453.
 Hal ini divisualisasikan pada garis-garis di BoxPlot.
 Jarak antarkuartil, IQR = Q3 - Q1, bernilai 1452.75, yaitu bagian kotak pada BoxPlot.
 Kolom ini memiliki rata-rata 4801.246.
 Kolom ini memiliki standar deviasi 986.3954914816017, variansi 972976.0656152305, dan range 4931.
 Kolom ini memiliki nilai skewness 0.23814408738280812, kurva positive skew (kurva condong ke kiri, sebagian besar nilai berada di nilai rendah).
 Kolom ini memiliki nilai kurtosis -0.4346305340273977, kurva platykurtic (nilai <3).
 Dari BoxPlot, dapat dilihat bahwa tidak ada outlier pada kolom Daerah.

In [209]:

```
print("SOAL 2: Boxplot Sumbu Utama")
%matplotlib inline
boxplot = df.boxplot(column=["SumbuUtama"])
```

SOAL 2: Boxplot Sumbu Utama



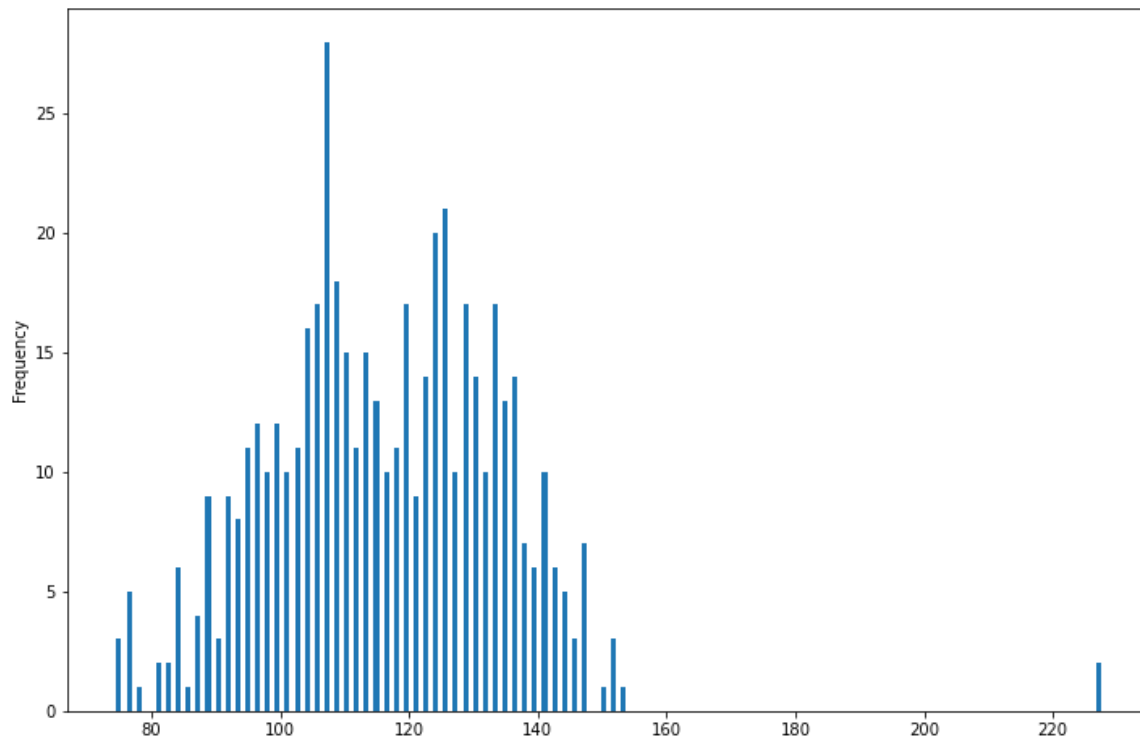
In [210]:

```
print("SOAL 2: Histogram Sumbu Utama")  
df["SumbuUtama"].plot.hist(bins=100,figsize=(12,8),rwidth=0.5)
```

SOAL 2: Histogram Sumbu Utama

Out[210]:

<matplotlib.axes._subplots.AxesSubplot at 0x7fd51174ca00>



In [211]:

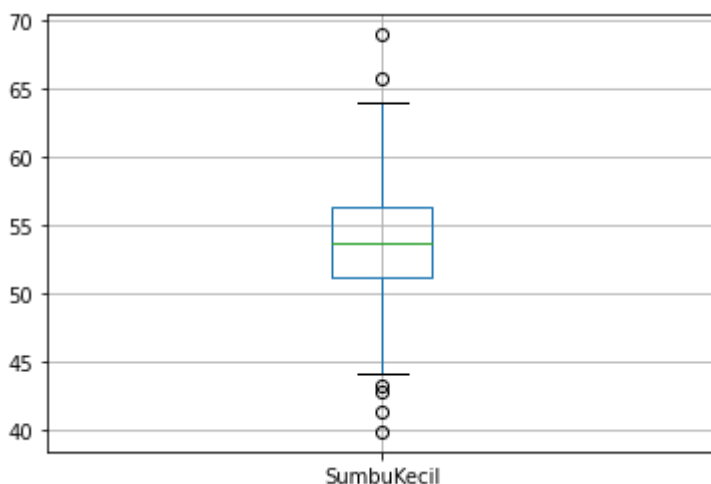
```
print("Kolom Sumbu Utama memiliki nilai minimum 74.13311404, Q1 104.116098, median 115.40513995, Q3 129.046792, dan nilai maksimum 227.9285827.")
print("Hal ini divisualisasikan pada garis-garis di BoxPlot.")
print("Jarak antarkuartil, IQR = Q3 - Q1, bernilai 24.930693849999997, yaitu bagian kotak pada BoxPlot.")
print("Kolom ini memiliki rata-rata 116.04517136778.")
print("Kolom ini memiliki standar deviasi 18.28262595755935, variansi 334.25441190402296, dan range 153.79546865999998.")
print("Kolom ini memiliki nilai skewness 0.7615287378076652, kurva positive skew (kurva condong ke kiri, sebagian besar nilai berada di nilai rendah).")
print("Kolom ini memiliki nilai kurtosis 4.33053354843697, kurva leptokurtic (nilai >3).")
print("Dari BoxPlot, dapat dilihat bahwa ada outlier pada kolom Sumbu Utama, yaitu di bagian ekor yang mengarah ke nilai maksimum.")
```

Kolom Sumbu Utama memiliki nilai minimum 74.13311404, Q1 104.116098, median 115.40513995, Q3 129.046792, dan nilai maksimum 227.9285827. Hal ini divisualisasikan pada garis-garis di BoxPlot. Jarak antarkuartil, IQR = Q3 - Q1, bernilai 24.930693849999997, yaitu bagian kotak pada BoxPlot. Kolom ini memiliki rata-rata 116.04517136778. Kolom ini memiliki standar deviasi 18.28262595755935, variansi 334.25441190402296, dan range 153.79546865999998. Kolom ini memiliki nilai skewness 0.7615287378076652, kurva positive skew (kurva condong ke kiri, sebagian besar nilai berada di nilai rendah). Kolom ini memiliki nilai kurtosis 4.33053354843697, kurva leptokurtic (nilai >3). Dari BoxPlot, dapat dilihat bahwa ada outlier pada kolom Sumbu Utama, yaitu di bagian ekor yang mengarah ke nilai maksimum.

In [212]:

```
print("SOAL 2: Boxplot Sumbu Kecil")
%matplotlib inline
boxplot = df.boxplot(column=["SumbuKecil"])
```

SOAL 2: Boxplot Sumbu Kecil



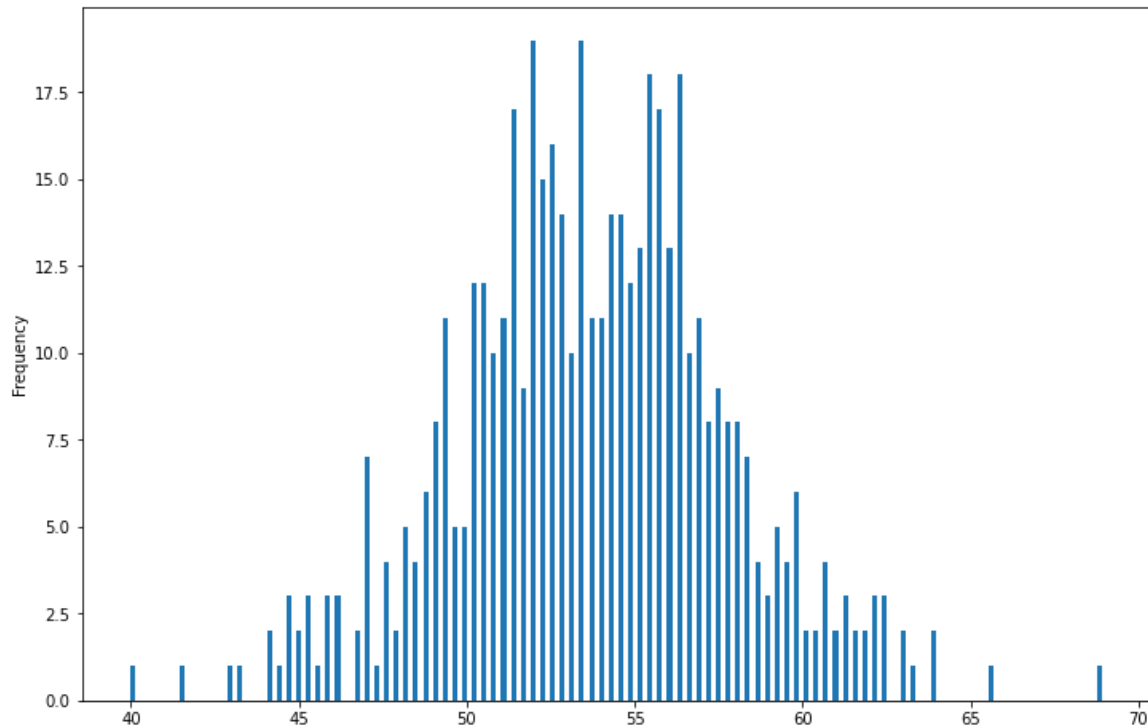
In [213]:

```
print("SOAL 2: Histogram Sumbu Kecil")  
df["SumbuKecil"].plot.hist(bins=100,figsize=(12,8),rwidth=0.5)
```

SOAL 2: Histogram Sumbu Kecil

Out[213]:

<matplotlib.axes._subplots.AxesSubplot at 0x7fd514d8e430>



In [214]:

```
print("Kolom Sumbu Kecil memiliki nilai minimum 39.90651744, Q1 51.193576, median 53.731198774999996, Q3 56.325158, dan nilai maksimum 68.97769987.")
print("Hal ini divisualisasikan pada garis-garis di BoxPlot.")
print("Jarak antarkuartil, IQR = Q3 - Q1, bernilai 5.131581650000001, yaitu bagian kotak pada BoxPlot.")
print("Kolom ini memiliki rata-rata 53.71524598896.")
print("Kolom ini memiliki standar deviasi 4.0710747524750355, variansi 16.57364964023967, dan range 29.071182429999993.")
print("Kolom ini memiliki nilai skewness -0.010828051555611359, kurva negative skew (kurva condong ke kanan, sebagian besar nilai berada di nilai tinggi).")
print("Kolom ini memiliki nilai kurtosis 0.475568450438137, kurva platykurtic (nilai <3).")
print("Dari BoxPlot, dapat dilihat bahwa ada outlier pada kolom Sumbu Kecil, yaitu di kedua bagian ekor, yang mengarah ke nilai minimum dan maksimum.")
```

Kolom Sumbu Kecil memiliki nilai minimum 39.90651744, Q1 51.193576, median 53.731198774999996, Q3 56.325158, dan nilai maksimum 68.97769987.

Hal ini divisualisasikan pada garis-garis di BoxPlot.

Jarak antarkuartil, IQR = Q3 - Q1, bernilai 5.131581650000001, yaitu bagian kotak pada BoxPlot.

Kolom ini memiliki rata-rata 53.71524598896.

Kolom ini memiliki standar deviasi 4.0710747524750355, variansi 16.57364964023967, dan range 29.071182429999993.

Kolom ini memiliki nilai skewness -0.010828051555611359, kurva negative skew (kurva condong ke kanan, sebagian besar nilai berada di nilai tinggi).

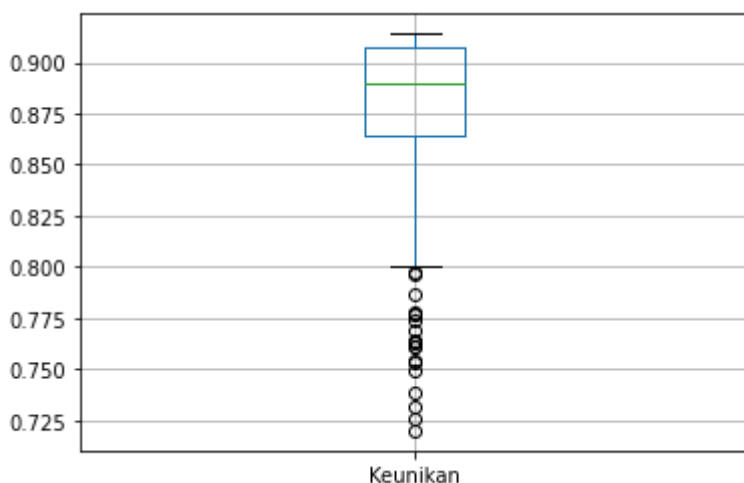
Kolom ini memiliki nilai kurtosis 0.475568450438137, kurva platykurtic (nilai <3).

Dari BoxPlot, dapat dilihat bahwa ada outlier pada kolom Sumbu Kecil, yaitu di kedua bagian ekor, yang mengarah ke nilai minimum dan maksimum.

In [215]:

```
print("SOAL 2: Boxplot Keunikan")
%matplotlib inline
boxplot = df.boxplot(column=["Keunikan"])
```

SOAL 2: Boxplot Keunikan



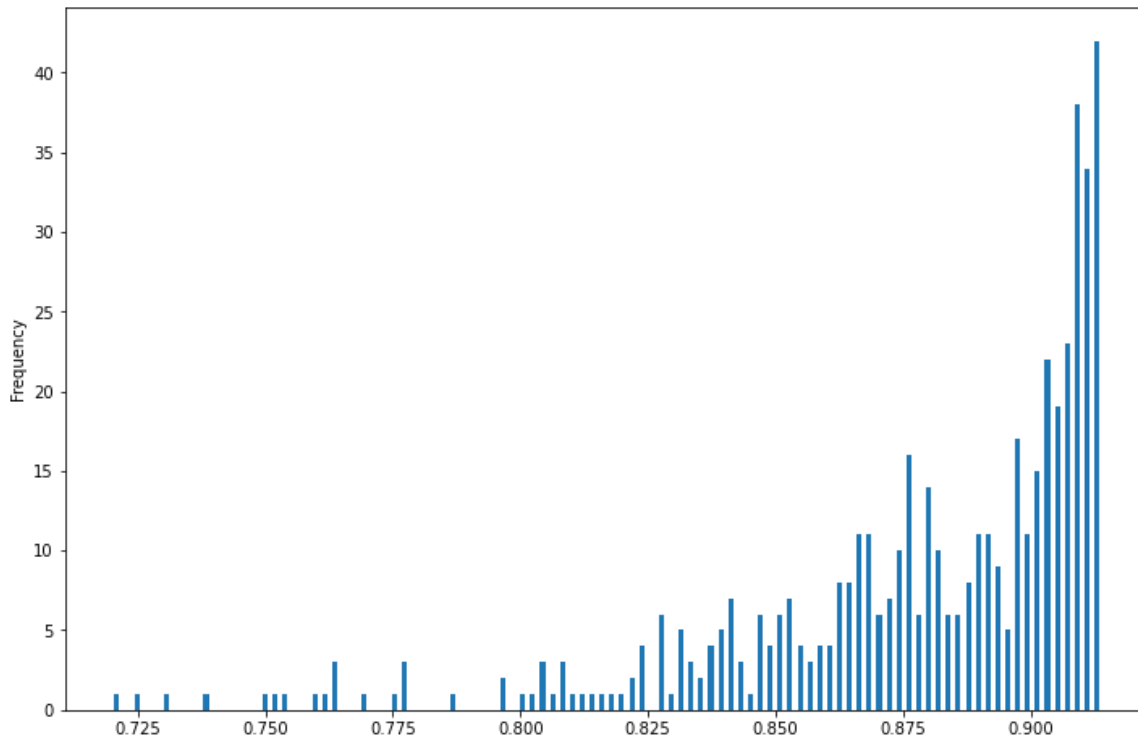
In [216]:

```
print("SOAL 2: Histogram Keunikan")  
df["Keunikan"].plot.hist(bins=100,figsize=(12,8),rwidth=0.5)
```

SOAL 2: Histogram Keunikan

Out[216]:

<matplotlib.axes._subplots.AxesSubplot at 0x7fd5150e82b0>



In [217]:

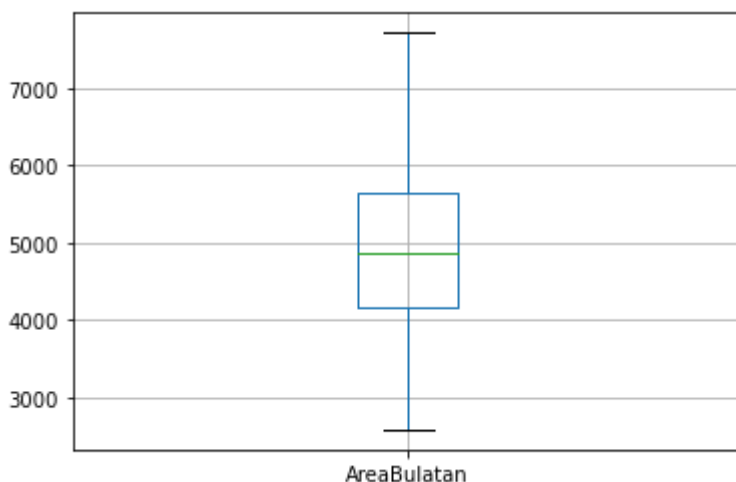
```
print("Kolom Keunikan memiliki nilai minimum 0.719916226, Q1 0.863676, median 0.8900454185, Q3 0.907578, dan nilai maksimum 0.914001406.")
print("Hal ini divisualisasikan pada garis-garis di BoxPlot.")
print("Jarak antarkuartil, IQR = Q3 - Q1, bernilai 0.04390216425000004, yaitu bagian kotak pada BoxPlot.")
print("Kolom ini memiliki rata-rata 0.8787639143399999.")
print("Kolom ini memiliki standar deviasi 0.036585905504777014, variansi 0.0013385284816044732, dan range 0.19408517999999997.")
print("Kolom ini memiliki nilai skewness -1.6234718222806501, kurva negative skew (kurva condong ke kanan, sebagian besar nilai berada di nilai tinggi).")
print("Kolom ini memiliki nilai kurtosis 2.917255925694391, kurva platykurtic (nilai <3).")
print("Dari BoxPlot, dapat dilihat bahwa ada outlier pada kolom Keunikan, yaitu di bagian ekor yang mengarah ke nilai minimum.")
```

Kolom Keunikan memiliki nilai minimum 0.719916226, Q1 0.863676, median 0.8900454185, Q3 0.907578, dan nilai maksimum 0.914001406.
 Hal ini divisualisasikan pada garis-garis di BoxPlot.
 Jarak antarkuartil, IQR = Q3 - Q1, bernilai 0.04390216425000004, yaitu bagian kotak pada BoxPlot.
 Kolom ini memiliki rata-rata 0.8787639143399999.
 Kolom ini memiliki standar deviasi 0.036585905504777014, variansi 0.0013385284816044732, dan range 0.19408517999999997.
 Kolom ini memiliki nilai skewness -1.6234718222806501, kurva negative skew (kurva condong ke kanan, sebagian besar nilai berada di nilai tinggi).
 Kolom ini memiliki nilai kurtosis 2.917255925694391, kurva platykurtic (nilai <3).
 Dari BoxPlot, dapat dilihat bahwa ada outlier pada kolom Keunikan, yaitu di bagian ekor yang mengarah ke nilai minimum.

In [218]:

```
print("SOAL 2: Boxplot Area Bulatan")
%matplotlib inline
boxplot = df.boxplot(column=["AreaBulatan"])
```

SOAL 2: Boxplot Area Bulatan



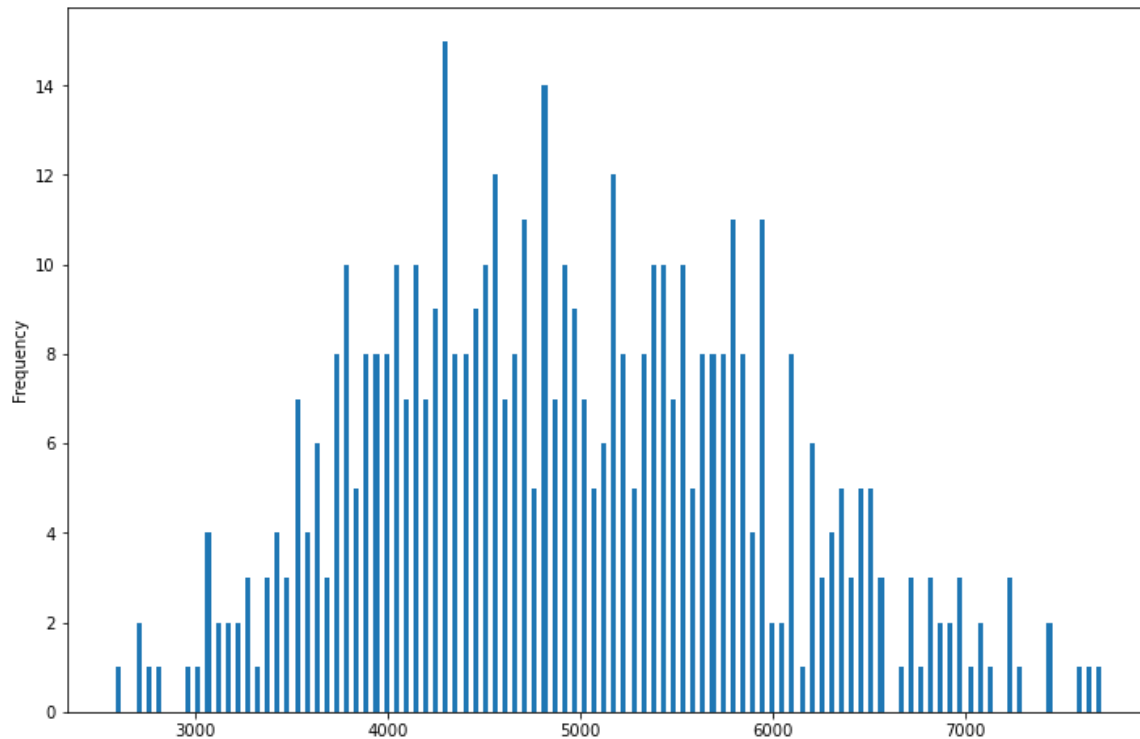
In [219]:

```
print("SOAL 2: Histogram Area Bulatan")  
df["AreaBulatan"].plot.hist(bins=100,figsize=(12,8),rwidth=0.5)
```

SOAL 2: Histogram Area Bulatan

Out[219]:

<matplotlib.axes._subplots.AxesSubplot at 0x7fd51543f8b0>



In [220]:

```
print("Kolom Area Bulatan memiliki nilai minimum 2579, Q1 4170.25, median 4857.0, Q3 5654.25, dan nilai maksimum 7720.")
print("Hal ini divisualisasikan pada garis-garis di BoxPlot.")
print("Jarak antarkuartil, IQR = Q3 - Q1, bernilai 1484.0, yaitu bagian kotak pada BoxPlot.")
print("Kolom ini memiliki rata-rata 4937.048.")
print("Kolom ini memiliki standar deviasi 1011.6962549701573, variansi 1023529.3123206414, dan range 5141.")
print("Kolom ini memiliki nilai skewness 0.2575600053152032, kurva positive skew (kurva condong ke kiri, sebagian besar nilai berada di nilai rendah).")
print("Kolom ini memiliki nilai kurtosis -0.40968492037366033, kurva platykurtic (nilai <3).")
print("Dari BoxPlot, dapat dilihat bahwa tidak ada outlier pada kolom Area Bulatan.")
```

Kolom Area Bulatan memiliki nilai minimum 2579, Q1 4170.25, median 4857.0, Q3 5654.25, dan nilai maksimum 7720.

Hal ini divisualisasikan pada garis-garis di BoxPlot.

Jarak antarkuartil, IQR = Q3 - Q1, bernilai 1484.0, yaitu bagian kotak pada BoxPlot.

Kolom ini memiliki rata-rata 4937.048.

Kolom ini memiliki standar deviasi 1011.6962549701573, variansi 1023529.3123206414, dan range 5141.

Kolom ini memiliki nilai skewness 0.2575600053152032, kurva positive skew (kurva condong ke kiri, sebagian besar nilai berada di nilai rendah).

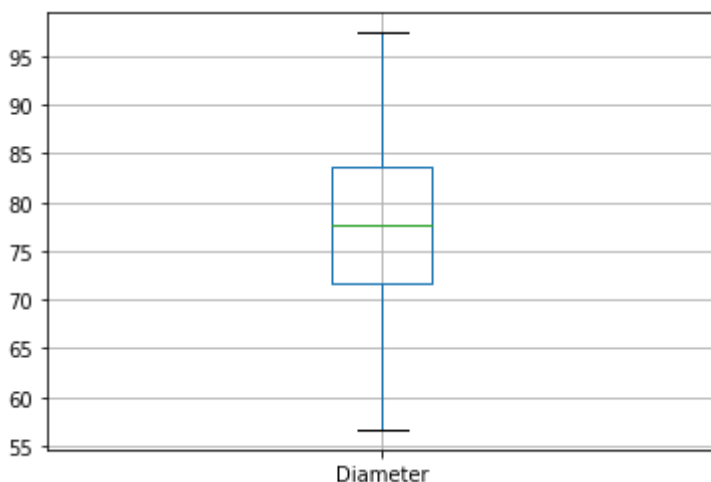
Kolom ini memiliki nilai kurtosis -0.40968492037366033, kurva platykurtic (nilai <3).

Dari BoxPlot, dapat dilihat bahwa tidak ada outlier pada kolom Area Bulatan.

In [221]:

```
print("SOAL 2: Boxplot Diameter")
%matplotlib inline
boxplot = df.boxplot(column=["Diameter"])
```

SOAL 2: Boxplot Diameter



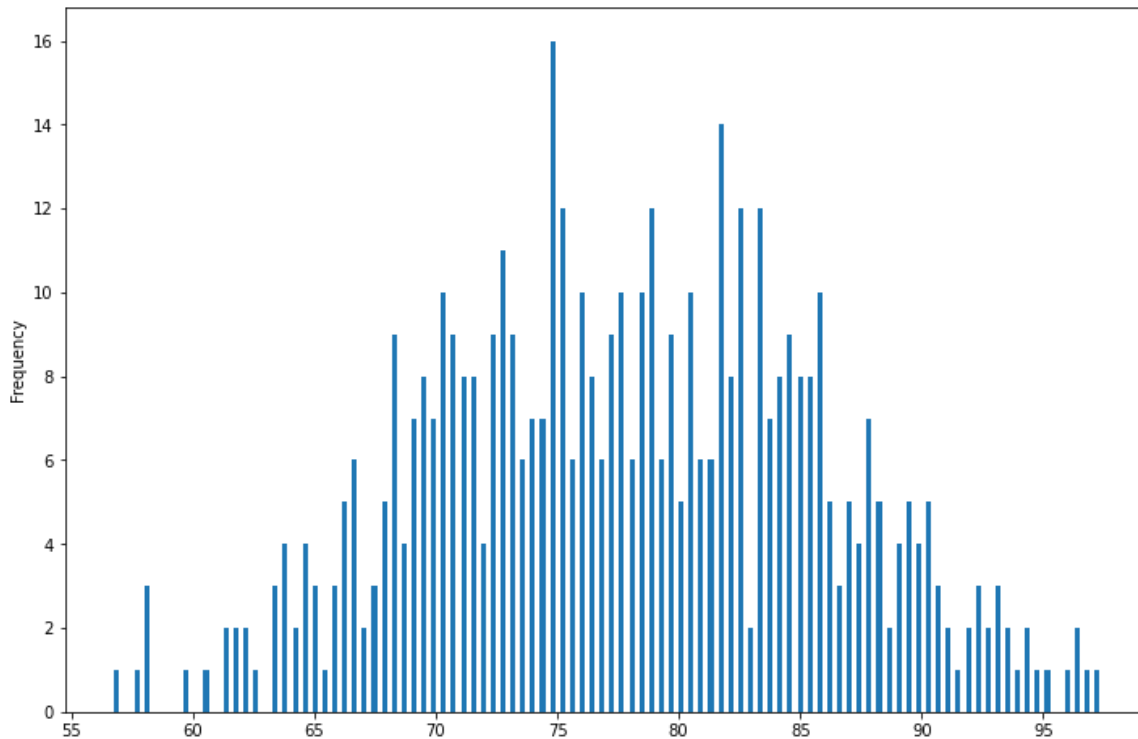
In [222]:

```
print("SOAL 2: Histogram Diameter")
df["Diameter"].plot.hist(bins=100,figsize=(12,8),rwidth=0.5)
```

SOAL 2: Histogram Diameter

Out[222]:

<matplotlib.axes._subplots.AxesSubplot at 0x7fd511f0c910>



In [223]:

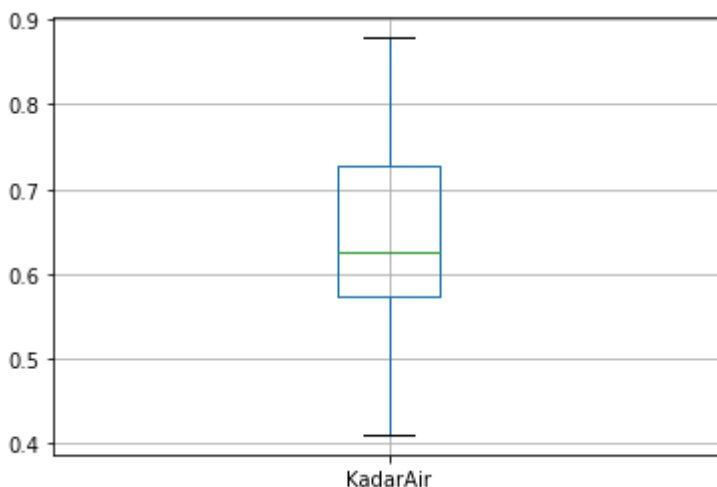
```
print("Kolom Diameter memiliki nilai minimum 56.66665803, Q1 71.745308, median 77.64527658, Q3 83.648598, dan nilai maksimum 97.41383027.")
print("Hal ini divisualisasikan pada garis-garis di BoxPlot.")
print("Jarak antarkuartil, IQR = Q3 - Q1, bernilai 11.90329002, yaitu bagian kotak pada BoxPlot.")
print("Kolom ini memiliki rata-rata 77.77115780832001.")
print("Kolom ini memiliki standar deviasi 8.056867291849713, variansi 64.91311055847774, dan range 40.747172240000005.")
print("Kolom ini memiliki nilai skewness 0.002724966865193717, kurva positive skew (kurva sedikit condong ke kiri, sebagian besar nilai berada di nilai rendah).")
print("Kolom ini memiliki nilai kurtosis -0.46645451303121455, kurva platykurtic (nilai <3).")
print("Dari BoxPlot, dapat dilihat bahwa tidak ada outlier pada kolom Diameter.")
```

Kolom Diameter memiliki nilai minimum 56.66665803, Q1 71.745308, median 77.64527658, Q3 83.648598, dan nilai maksimum 97.41383027.
 Hal ini divisualisasikan pada garis-garis di BoxPlot.
 Jarak antarkuartil, IQR = Q3 - Q1, bernilai 11.90329002, yaitu bagian kotak pada BoxPlot.
 Kolom ini memiliki rata-rata 77.77115780832001.
 Kolom ini memiliki standar deviasi 8.056867291849713, variansi 64.91311055847774, dan range 40.747172240000005.
 Kolom ini memiliki nilai skewness 0.002724966865193717, kurva positive skew (kurva sedikit condong ke kiri, sebagian besar nilai berada di nilai rendah).
 Kolom ini memiliki nilai kurtosis -0.46645451303121455, kurva platykurtic (nilai <3).
 Dari BoxPlot, dapat dilihat bahwa tidak ada outlier pada kolom Diameter.

In [224]:

```
print("SOAL 2: Boxplot Kadar Air")
%matplotlib inline
boxplot = df.boxplot(column=["KadarAir"])
```

SOAL 2: Boxplot Kadar Air



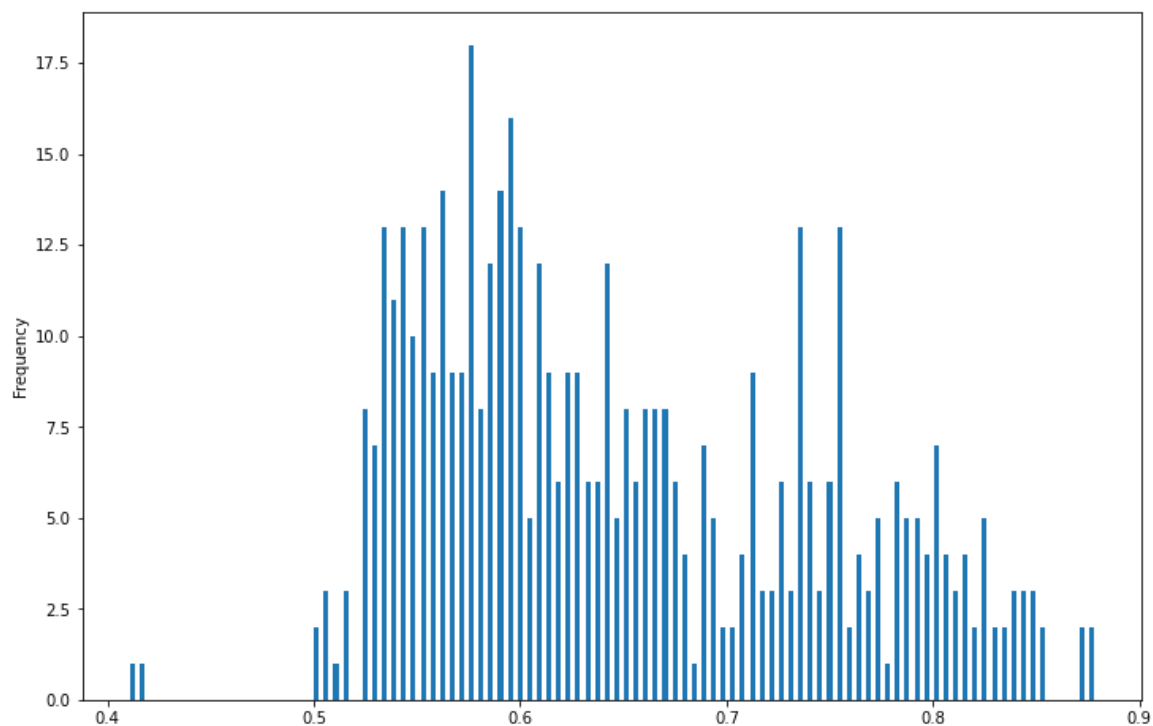
In [225]:

```
print("SOAL 2: Histogram Kadar Air")
df["KadarAir"].plot.hist(bins=100,figsize=(12,8),rwidth=0.5)
```

SOAL 2: Histogram Kadar Air

Out[225]:

<matplotlib.axes._subplots.AxesSubplot at 0x7fd52ee15340>



In [226]:

```
print("Kolom Kadar Air memiliki nilai minimum 0.409927152, Q1 0.572632, median
      0.626116699, Q3 0.726633, dan nilai maksimum 0.878899083.")
print("Hal ini divisualisasikan pada garis-garis di BoxPlot.")
print("Jarak antarkuartil, IQR = Q3 - Q1, bernilai 0.15400088724999994, yaitu ba
      gian kotak pada BoxPlot.")
print("Kolom ini memiliki rata-rata 0.6483716718979999.")
print("Kolom ini memiliki standar deviasi 0.09436709809379533, variansi 0.008905
      14920264399, dan range 0.46897193099999995.")
print("Kolom ini memiliki nilai skewness 0.49366131797330265, kurva positive ske
      w (kurva condong ke kiri, sebagian besar nilai berada di nilai rendah).")
print("Kolom ini memiliki nilai kurtosis -0.7403261705867821, kurva platykurtic
      (nilai <3).")
print("Dari BoxPlot, dapat dilihat bahwa tidak ada outlier pada kolom Kadar Ai
      r.")
```

Kolom Kadar Air memiliki nilai minimum 0.409927152, Q1 0.572632, median 0.626116699, Q3 0.726633, dan nilai maksimum 0.878899083.

Hal ini divisualisasikan pada garis-garis di BoxPlot.

Jarak antarkuartil, IQR = Q3 - Q1, bernilai 0.15400088724999994, yaitu bagian kotak pada BoxPlot.

Kolom ini memiliki rata-rata 0.6483716718979999.

Kolom ini memiliki standar deviasi 0.09436709809379533, variansi 0.00890514920264399, dan range 0.46897193099999995.

Kolom ini memiliki nilai skewness 0.49366131797330265, kurva positive skew (kurva condong ke kiri, sebagian besar nilai berada di nilai rendah).

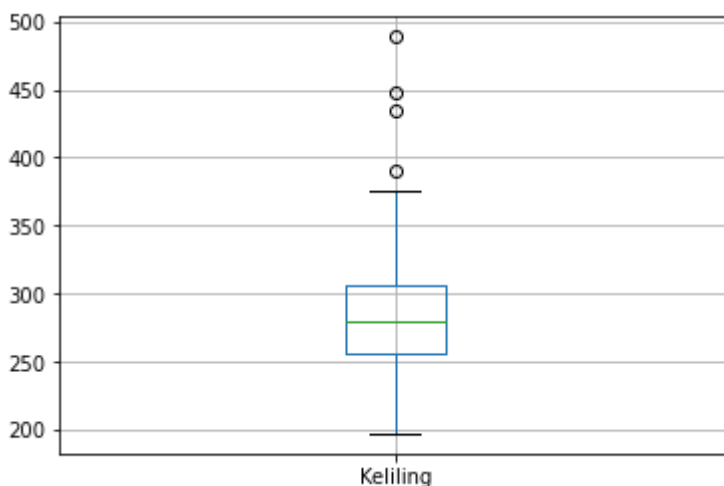
Kolom ini memiliki nilai kurtosis -0.7403261705867821, kurva platykurtic (nilai <3).

Dari BoxPlot, dapat dilihat bahwa tidak ada outlier pada kolom Kadar Air.

In [227]:

```
print("SOAL 2: Boxplot Keliling")
%matplotlib inline
boxplot = df.boxplot(column=["Keliling"])
```

SOAL 2: Boxplot Keliling



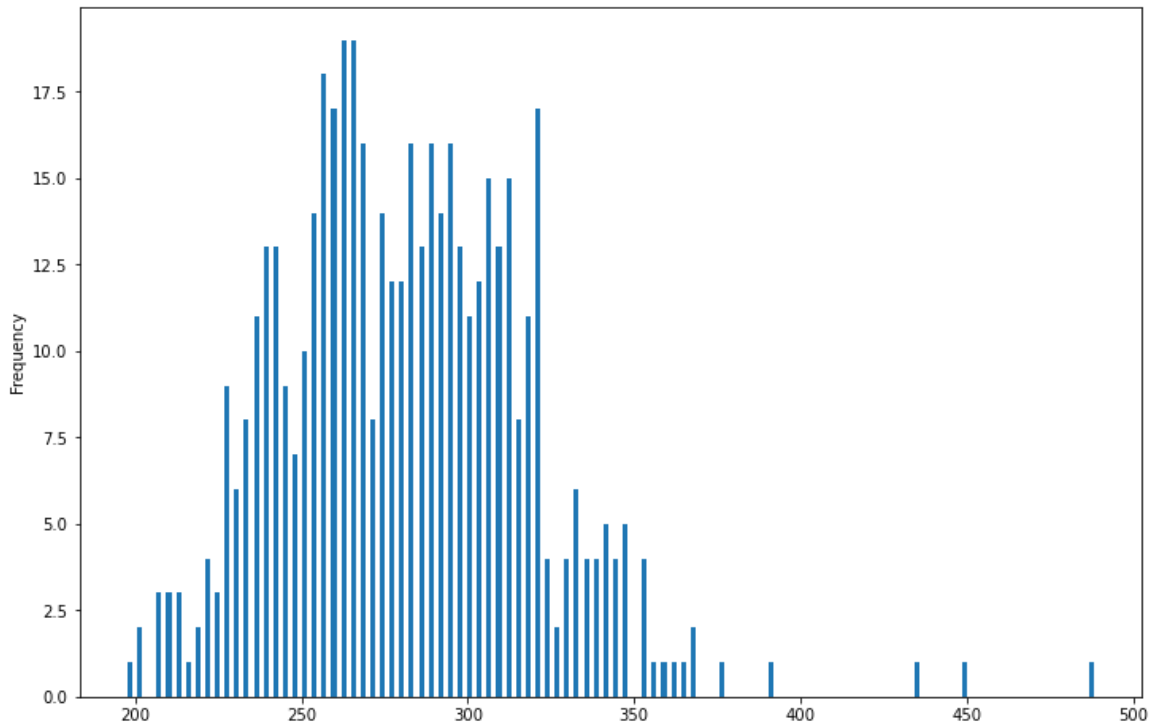
In [228]:

```
print("SOAL 2: Histogram Keliling")  
df["Keliling"].plot.hist(bins=100,figsize=(12,8),rwidth=0.5)
```

SOAL 2: Histogram Keliling

Out[228]:

<matplotlib.axes._subplots.AxesSubplot at 0x7fd5144c5850>



In [229]:

```
print("Kolom Keliling memiliki nilai minimum 197.015, Q1 255.8830, median 280.0455, Q3 306.0625, dan nilai maksimum 488.837.")
print("Hal ini divisualisasikan pada garis-garis di BoxPlot.")
print("Jarak antarkuartil, IQR = Q3 - Q1, bernilai 50.17949999999999, yaitu bagian kotak pada BoxPlot.")
print("Kolom ini memiliki rata-rata 281.47972199999999.")
print("Kolom ini memiliki standar deviasi 37.33540171514401, variansi 1393.9322212311781, dan range 291.822.")
print("Kolom ini memiliki nilai skewness 0.7336269072005543, kurva positive skew (kurva condong ke kiri, sebagian besar nilai berada di nilai rendah).")
print("Kolom ini memiliki nilai kurtosis 2.272684731245573, kurva platykurtic (nilai <3).")
print("Dari BoxPlot, dapat dilihat bahwa ada outlier pada kolom Keliling, yaitu di bagian ekor yang mengarah ke nilai maksimum.")
```

Kolom Keliling memiliki nilai minimum 197.015, Q1 255.8830, median 280.0455, Q3 306.0625, dan nilai maksimum 488.837.

Hal ini divisualisasikan pada garis-garis di BoxPlot.

Jarak antarkuartil, IQR = Q3 - Q1, bernilai 50.17949999999999, yaitu bagian kotak pada BoxPlot.

Kolom ini memiliki rata-rata 281.47972199999999.

Kolom ini memiliki standar deviasi 37.33540171514401, variansi 1393.9322212311781, dan range 291.822.

Kolom ini memiliki nilai skewness 0.7336269072005543, kurva positive skew (kurva condong ke kiri, sebagian besar nilai berada di nilai rendah).

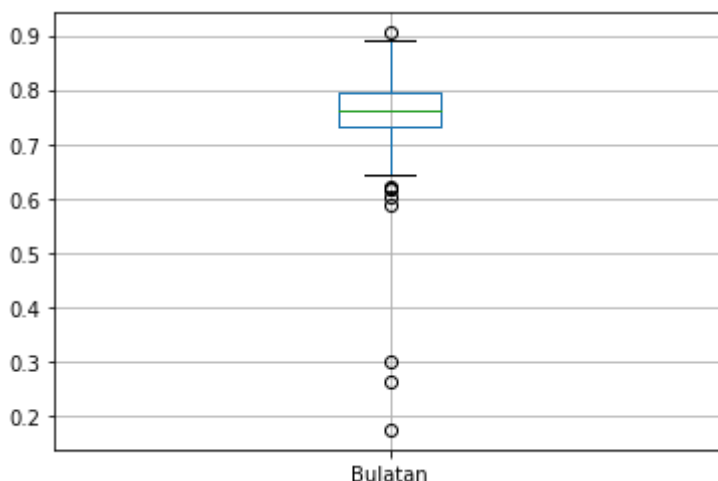
Kolom ini memiliki nilai kurtosis 2.272684731245573, kurva platykurtic (nilai <3).

Dari BoxPlot, dapat dilihat bahwa ada outlier pada kolom Keliling, yaitu di bagian ekor yang mengarah ke nilai maksimum.

In [230]:

```
print("SOAL 2: Boxplot Bulatan")
%matplotlib inline
boxplot = df.boxplot(column=["Bulatan"])
```

SOAL 2: Boxplot Bulatan



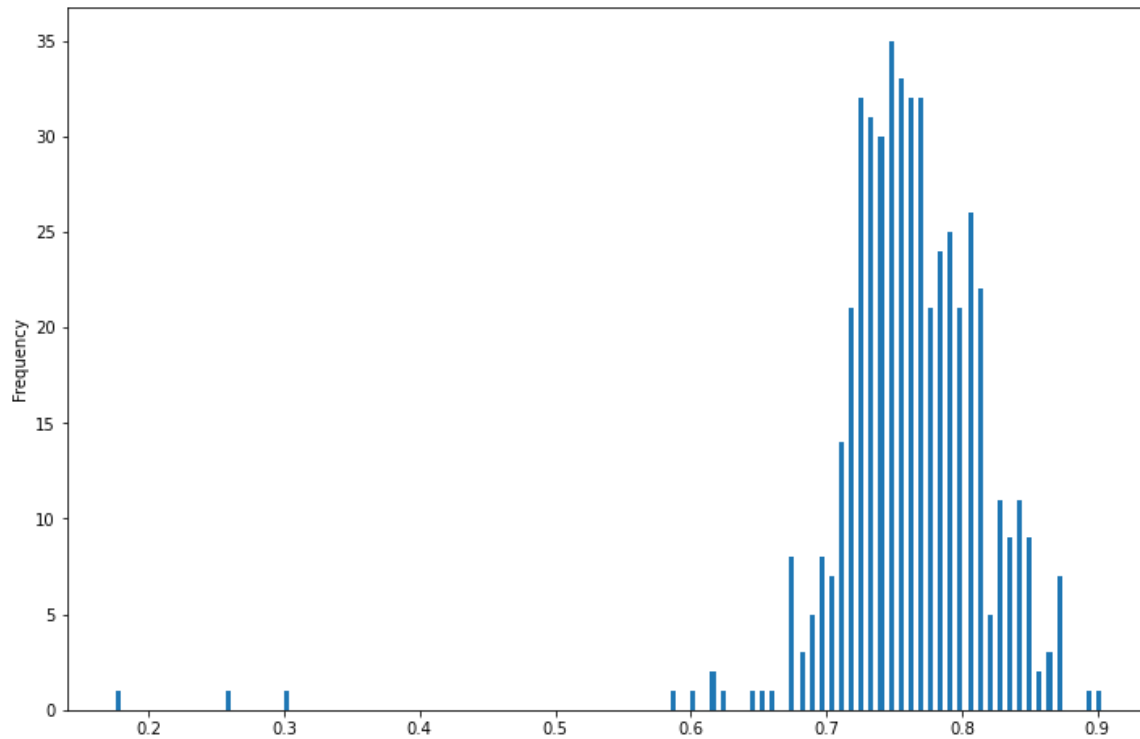
In [231]:

```
print("SOAL 2: Histogram Bulatan")  
df["Bulatan"].plot.hist(bins=100,figsize=(12,8),rwidth=0.5)
```

SOAL 2: Histogram Bulatan

Out[231]:

<matplotlib.axes._subplots.AxesSubplot at 0x7fd513d876a0>



In [232]:

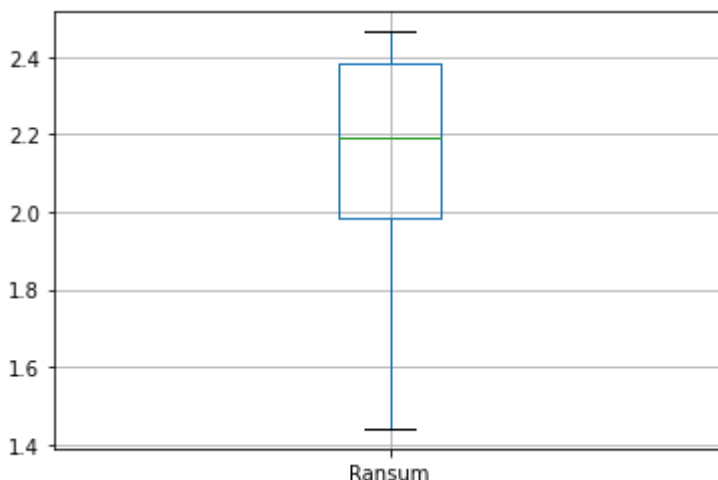
```
print("Kolom Bulatan memiliki nilai minimum 0.174590178, Q1 0.731991, median 0.7612884394999999, Q3 0.796361, dan nilai maksimum 0.904748313.")
print("Hal ini divisualisasikan pada garis-garis di BoxPlot.")
print("Jarak antarkuartil, IQR = Q3 - Q1, bernilai 0.06437024175000006, yaitu bagian kotak pada BoxPlot.")
print("Kolom ini memiliki rata-rata 0.7617374363080001.")
print("Kolom ini memiliki standar deviasi 0.06170246078673261, variansi 0.0038071936671382756, dan range 0.730158135.")
print("Kolom ini memiliki nilai skewness -3.599236766361642, kurva negative skew (kurva condong ke kanan, sebagian besar nilai berada di nilai tinggi).")
print("Kolom ini memiliki nilai kurtosis 29.975095904630063, kurva leptokurtic (nilai >3).")
print("Dari BoxPlot, dapat dilihat bahwa ada outlier pada kolom Bulatan, yaitu di kedua bagian ekor, yang mengarah ke nilai minimum dan maksimum.")
```

Kolom Bulatan memiliki nilai minimum 0.174590178, Q1 0.731991, median 0.7612884394999999, Q3 0.796361, dan nilai maksimum 0.904748313. Hal ini divisualisasikan pada garis-garis di BoxPlot. Jarak antarkuartil, IQR = Q3 - Q1, bernilai 0.06437024175000006, yaitu bagian kotak pada BoxPlot. Kolom ini memiliki rata-rata 0.7617374363080001. Kolom ini memiliki standar deviasi 0.06170246078673261, variansi 0.0038071936671382756, dan range 0.730158135. Kolom ini memiliki nilai skewness -3.599236766361642, kurva negative skew (kurva condong ke kanan, sebagian besar nilai berada di nilai tinggi). Kolom ini memiliki nilai kurtosis 29.975095904630063, kurva leptokurtic (nilai >3). Dari BoxPlot, dapat dilihat bahwa ada outlier pada kolom Bulatan, yaitu di kedua bagian ekor, yang mengarah ke nilai minimum dan maksimum.

In [233]:

```
print("SOAL 2: Boxplot Ransum")
%matplotlib inline
boxplot = df.boxplot(column=["Ransum"])
```

SOAL 2: Boxplot Ransum



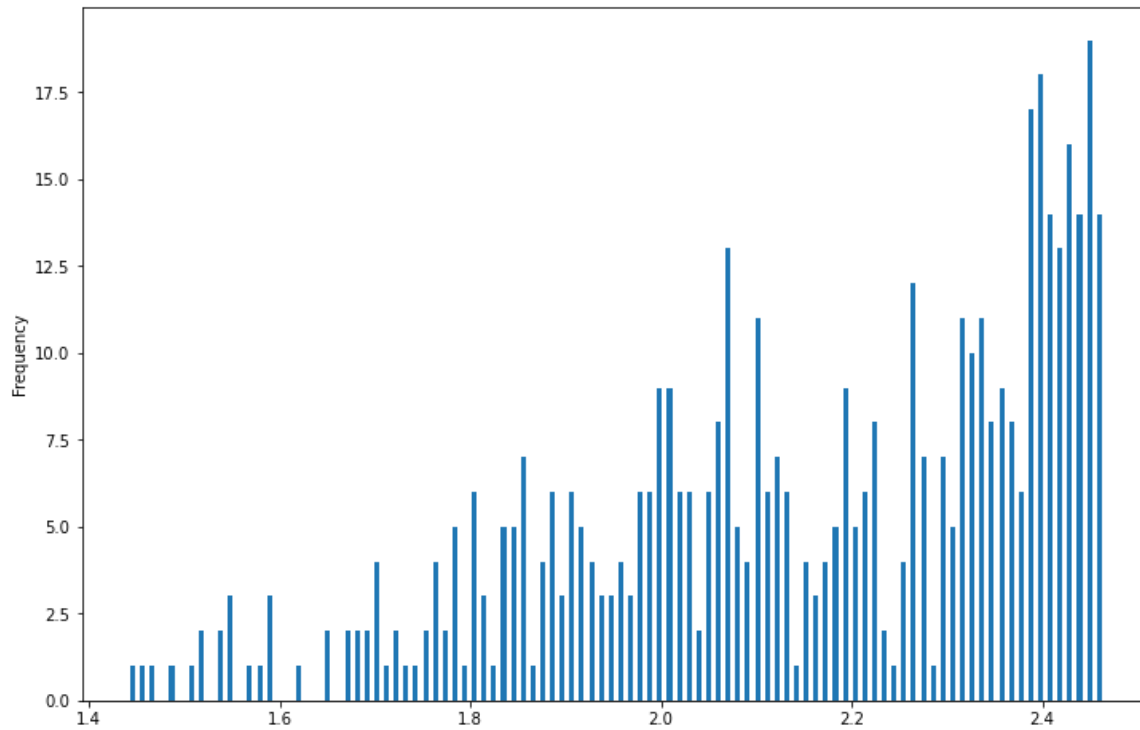
In [234]:

```
print("SOAL 2: Histogram Ransum")  
df["Ransum"].plot.hist(bins=100,figsize=(12,8),rwidth=0.5)
```

SOAL 2: Histogram Ransum

Out[234]:

<matplotlib.axes._subplots.AxesSubplot at 0x7fd514305370>



In [235]:

```
print("Kolom Ransum memiliki nilai minimum 1.440795615, Q1 1.983939, median 2.1935990365, Q3 2.381612, dan nilai maksimum 2.464808581.")
print("Hal ini divisualisasikan pada garis-garis di BoxPlot.")
print("Jarak antarkuartil, IQR = Q3 - Q1, bernilai 0.3976734275, yaitu bagian kotak pada BoxPlot.")
print("Kolom ini memiliki rata-rata 2.150915331084.")
print("Kolom ini memiliki standar deviasi 0.2497668892706582, variansi 0.06238349897594124, dan range 1.0240129660000001.")
print("Kolom ini memiliki nilai skewness -0.6581880925333655, kurva negative skew (kurva condong ke kanan, sebagian besar nilai berada di nilai tinggi).")
print("Kolom ini memiliki nilai kurtosis -0.4286557930626147, kurva platykurtic (nilai <3).")
print("Dari BoxPlot, dapat dilihat bahwa tidak ada outlier pada kolom Ransum.")
```

Kolom Ransum memiliki nilai minimum 1.440795615, Q1 1.983939, median 2.1935990365, Q3 2.381612, dan nilai maksimum 2.464808581.

Hal ini divisualisasikan pada garis-garis di BoxPlot.

Jarak antarkuartil, IQR = Q3 - Q1, bernilai 0.3976734275, yaitu bagian kotak pada BoxPlot.

Kolom ini memiliki rata-rata 2.150915331084.

Kolom ini memiliki standar deviasi 0.2497668892706582, variansi 0.06238349897594124, dan range 1.0240129660000001.

Kolom ini memiliki nilai skewness -0.6581880925333655, kurva negative skew (kurva condong ke kanan, sebagian besar nilai berada di nilai tinggi).

Kolom ini memiliki nilai kurtosis -0.4286557930626147, kurva platykurtic (nilai <3).

Dari BoxPlot, dapat dilihat bahwa tidak ada outlier pada kolom Ransum.

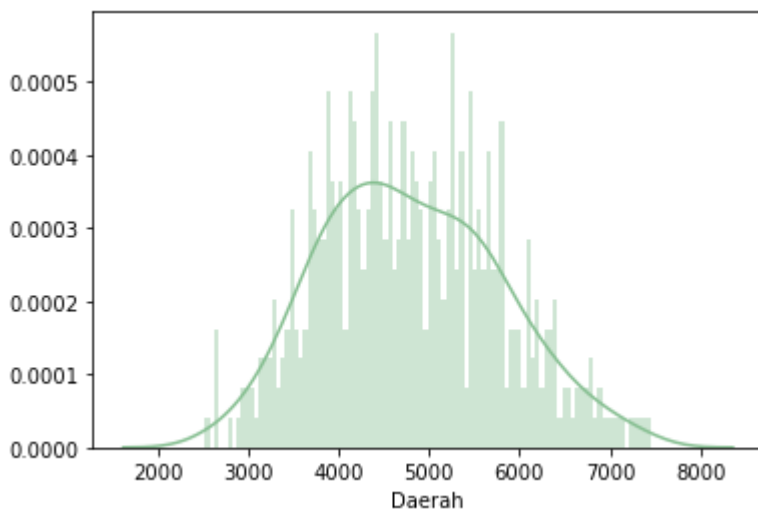
In [236]:

```
print("SOAL 3: Normality test Daerah")
sns.distplot(df['Daerah'],color='#86bf91', bins=100)
stat, p = normaltest(df["Daerah"])
print("stat=%.3f, p=%.3f" % (stat, p))
if (p > 0.05):
    print("Terdistribusi normal karena nilai p lebih besar dari 0.05 dan histogr  
am simetris")
else:
    print("Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan h  
istogram tidak simetris")
```

SOAL 3: Normality test Daerah

stat=10.859, p=0.004

Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan
histogram tidak simetris



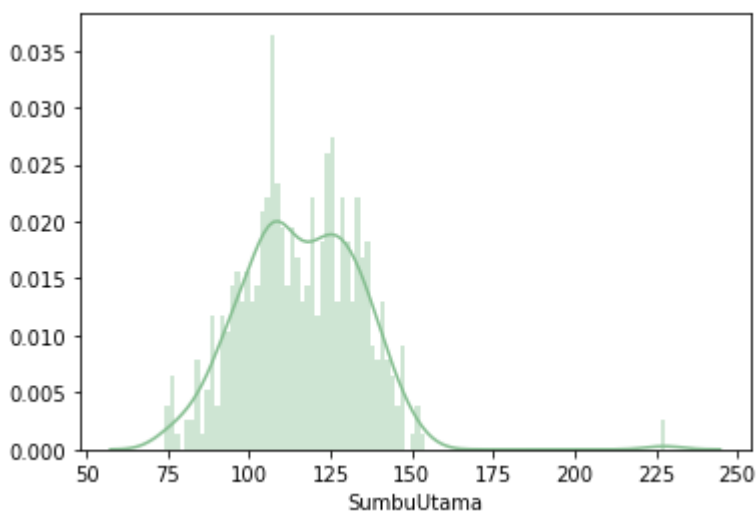
In [237]:

```
print("SOAL 3: Normality test Sumbu Utama")
sns.distplot(df['SumbuUtama'],color='#86bf91', bins=100)
stat, p = normaltest(df["SumbuUtama"])
print("stat=%.3f, p=%.3f" % (stat, p))
if (p > 0.05):
    print("Terdistribusi normal karena nilai p lebih besar dari 0.05 dan histogr  
am simetris")
else:
    print("Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan h  
istogram tidak simetris")
```

SOAL 3: Normality test Sumbu Utama

stat=95.120, p=0.000

Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan
histogram tidak simetris



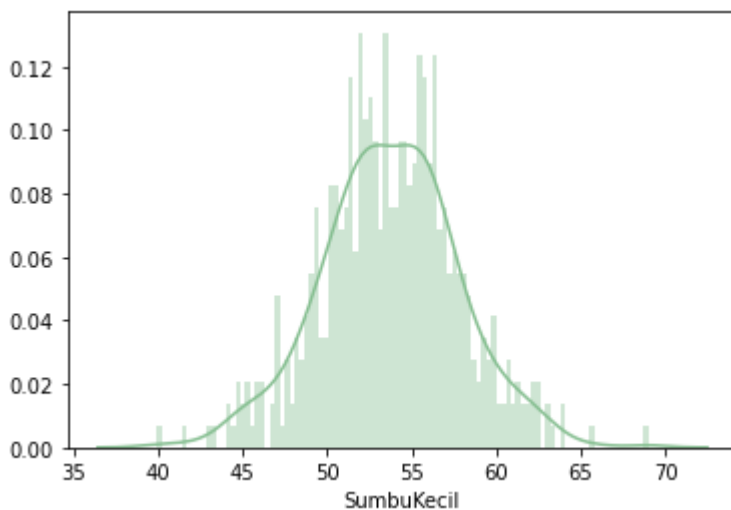
In [238]:

```
print("SOAL 3: Normality test Sumbu Kecil")
sns.distplot(df['SumbuKecil'],color='#86bf91', bins=100)
stat, p = normaltest(df["SumbuKecil"])
print("stat=%.3f, p=%.3f" % (stat, p))
if (p > 0.05):
    print("Terdistribusi normal karena nilai p lebih besar dari 0.05 dan histogr
am simetris")
else:
    print("Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan h
istogram tidak simetris")
```

SOAL 3: Normality test Sumbu Kecil

stat=3.698, p=0.157

Terdistribusi normal karena nilai p lebih besar dari 0.05 dan histog
ram simetris



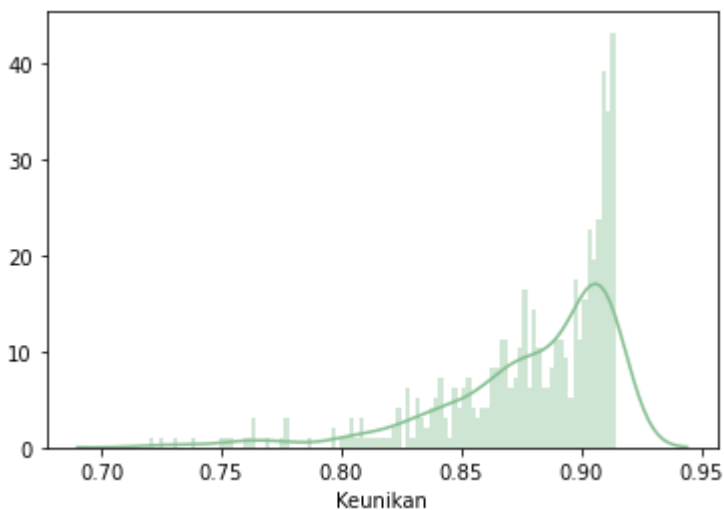
In [239]:

```
print("SOAL 3: Normality test Keunikan")
sns.distplot(df['Keunikan'],color='#86bf91', bins=100)
stat, p = normaltest(df["Keunikan"])
print("stat=%.3f, p=%.3f" % (stat, p))
if (p > 0.05):
    print("Terdistribusi normal karena nilai p lebih besar dari 0.05 dan histogr  
am simetris")
else:
    print("Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan h  
istogram tidak simetris")
```

SOAL 3: Normality test Keunikan

stat=158.617, p=0.000

Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan
histogram tidak simetris



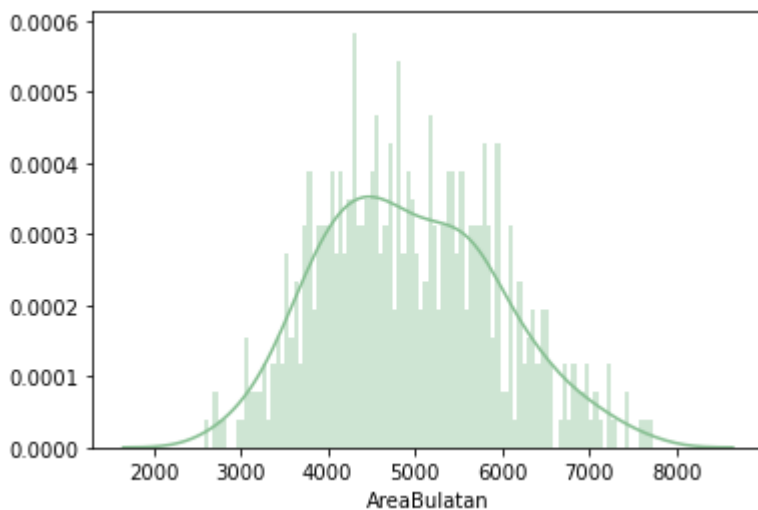
In [240]:

```
print("SOAL 3: Normality test Area Bulatan")
sns.distplot(df['AreaBulatan'],color='#86bf91', bins=100)
stat, p = normaltest(df["AreaBulatan"])
print("stat=%.3f, p=%.3f" % (stat, p))
if (p > 0.05):
    print("Terdistribusi normal karena nilai p lebih besar dari 0.05 dan histogr  
am simetris")
else:
    print("Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan h  
istogram tidak simetris")
```

SOAL 3: Normality test Area Bulatan

stat=10.739, p=0.005

Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan
histogram tidak simetris



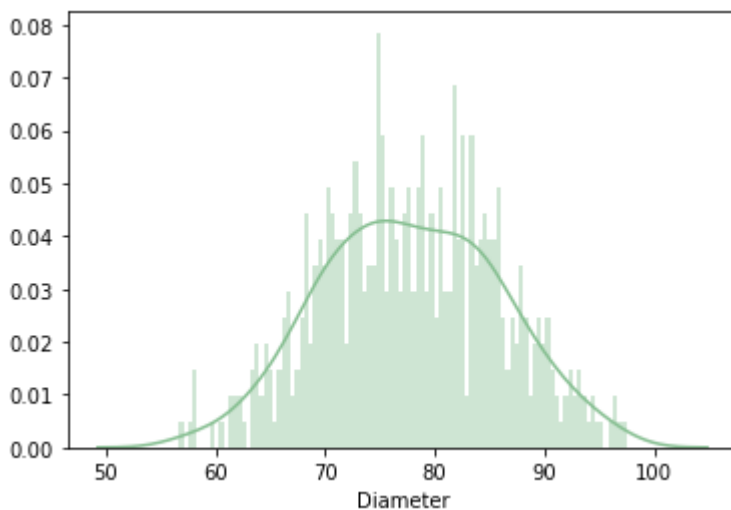
In [241]:

```
print("SOAL 3: Normality test Diameter")
sns.distplot(df['Diameter'],color='#86bf91', bins=100)
stat, p = normaltest(df["Diameter"])
print("stat=%.3f, p=%.3f" % (stat, p))
if (p > 0.05):
    print("Terdistribusi normal karena nilai p lebih besar dari 0.05 dan histogr
am simetris")
else:
    print("Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan h
istogram tidak simetris")
```

SOAL 3: Normality test Diameter

stat=7.446, p=0.024

Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan histogram tidak simetris



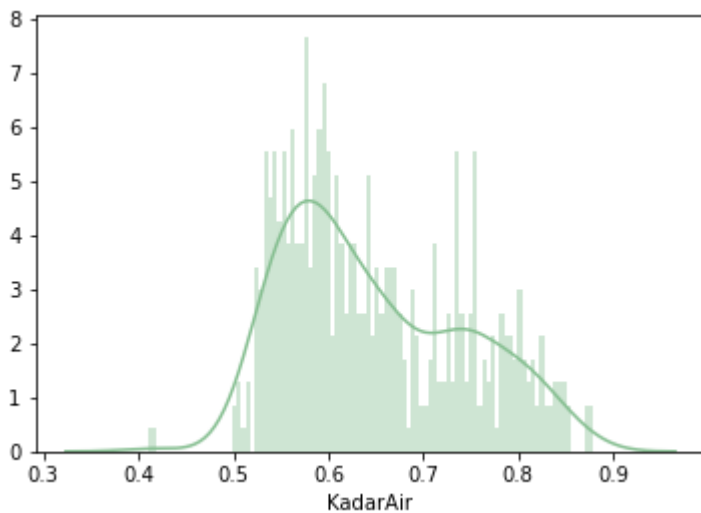
In [242]:

```
print("SOAL 3: Normality test Kadar Air")
sns.distplot(df['KadarAir'],color='#86bf91', bins=100)
stat, p = normaltest(df["KadarAir"])
print("stat=%.3f, p=%.3f" % (stat, p))
if (p > 0.05):
    print("Terdistribusi normal karena nilai p lebih besar dari 0.05 dan histogr  
am simetris")
else:
    print("Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan h  
istogram tidak simetris")
```

SOAL 3: Normality test Kadar Air

stat=48.756, p=0.000

Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan
histogram tidak simetris



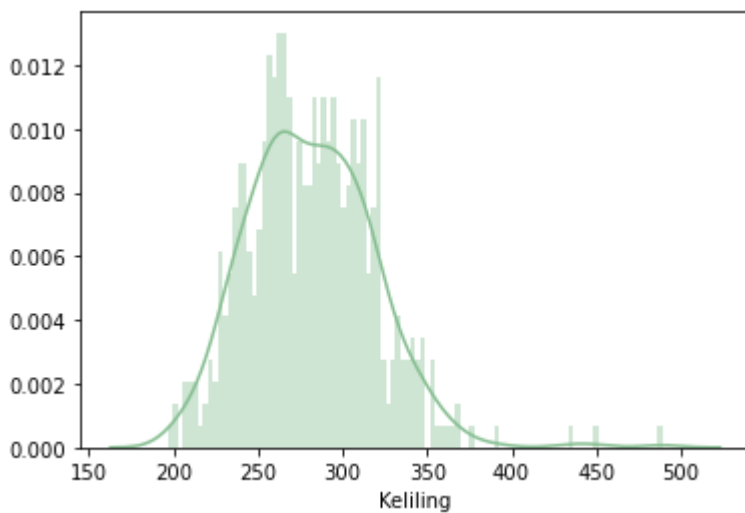
In [243]:

```
print("SOAL 3: Normality test Keliling")
sns.distplot(df['Keliling'],color='#86bf91', bins=100)
stat, p = normaltest(df["Keliling"])
print("stat=%.3f, p=%.3f" % (stat, p))
if (p > 0.05):
    print("Terdistribusi normal karena nilai p lebih besar dari 0.05 dan histogr  
am simetris")
else:
    print("Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan h  
istogram tidak simetris")
```

SOAL 3: Normality test Keliling

stat=67.465, p=0.000

Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan
histogram tidak simetris



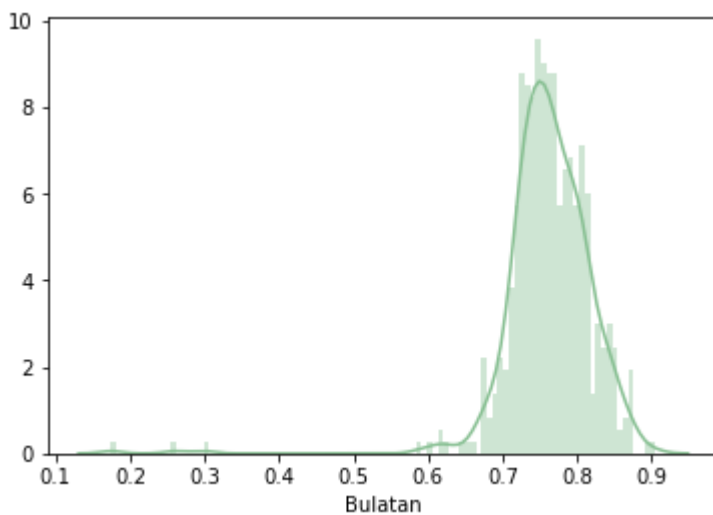
In [244]:

```
print("SOAL 3: Normality test Bulatan")
sns.distplot(df['Bulatan'],color='#86bf91', bins=100)
stat, p = normaltest(df["Bulatan"])
print("stat=%.3f, p=%.3f" % (stat, p))
if (p > 0.05):
    print("Terdistribusi normal karena nilai p lebih besar dari 0.05 dan histogr  
am simetris")
else:
    print("Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan h  
istogram tidak simetris")
```

SOAL 3: Normality test Bulatan

stat=442.119, p=0.000

Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan
histogram tidak simetris



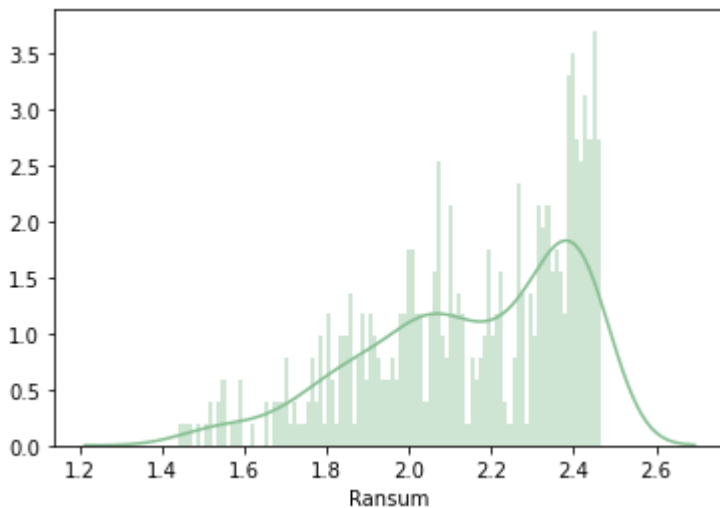
In [245]:

```
print("SOAL 3: Normality test Ransum")
sns.distplot(df['Ransum'],color='#86bf91', bins=100)
stat, p = normaltest(df["Ransum"])
print("stat=%.3f, p=%.3f" % (stat, p))
if (p > 0.05):
    print("Terdistribusi normal karena nilai p lebih besar dari 0.05 dan histogr  
am simetris")
else:
    print("Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan h  
istogram tidak simetris")
```

SOAL 3: Normality test Ransum

stat=37.060, p=0.000

Tidak terdistribusi normal karena nilai p lebih kecil dari 0.05 dan
histogram tidak simetris



In [246]:

```
print("SOAL 4A")
print("Nilai rata-rata Daerah di atas 4700?\n")
print("Enam langkah testing:")
print("1. Tentukan hipotesis nol atau H0")
print("    H0:  $\mu = 4700$ ")
print("2. Tentukan hipotesis alternatif atau H1")
print("    H1:  $\mu > 4700$ ")
print("3. Tentukan tingkat signifikan  $\alpha$ ")
sig = 0.05
print("     $\alpha =$ ", sig)
print("4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis")
daerahKritis = stats.norm.ppf(1-sig)
print("    Didapatkan daerah kritis  $z >$ ", daerahKritis)
print("    Digunakan uji statistik  $z$ ")
print("5. Hitung uji nilai statistik dari data sample")
ztest ,pval = stats.ztest(df["Daerah"], x2=None, value=4700,alternative="large
r")
print("    p value = ", float(pval))
print("    z value = ", ztest)
print("6. Ambil keputusan apakah H0 ditolak atau tidak")
if (ztest > daerahKritis):
    print("    H0 ditolak karena nilai ztest berada pada daerah kritis")
else:
    print("    H0 diterima karena nilai ztest tidak berada pada daerah kritis")
if (pval < 0.05):
    print("    H0 ditolak karena p value lebih kecil dari nilai signifikan")
else:
    print("    H0 diterima karena p value lebih besar dari nilai signifikan")
print("Boxplot:")
%matplotlib inline
boxplot = df.boxplot(column=["Daerah"])
```

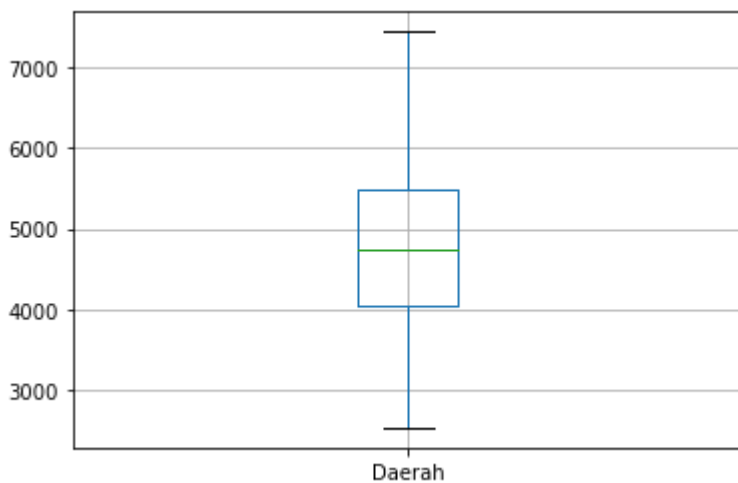

SOAL 4A

Nilai rata-rata Daerah di atas 4700?

Enam langkah testing:

1. Tentukan hipotesis nol atau H_0
 $H_0: \mu = 4700$
2. Tentukan hipotesis alternatif atau H_1
 $H_1: \mu > 4700$
3. Tentukan tingkat signifikan α
 $\alpha = 0.05$
4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis
Didapatkan daerah kritis $z > 1.6448536269514722$
Digunakan uji statistik z
5. Hitung uji nilai statistik dari data sample
 $p \text{ value} = 0.010862155196799894$
 $z \text{ value} = 2.295153824252517$
6. Ambil keputusan apakah H_0 ditolak atau tidak
 H_0 ditolak karena nilai z test berada pada daerah kritis
 H_0 ditolak karena p value lebih kecil dari nilai signifikan

Boxplot:



In [247]:

```
print("SOAL 4B")
print("Nilai Rata-rata Sumbu Utama tidak sama dengan 116?\n")
print("Enam langkah testing:")
print("1. Tentukan hipotesis nol atau H0")
print("    H0:  $\mu = 116$ ")
print("2. Tentukan hipotesis alternatif atau H1")
print("    H1:  $\mu \neq 116$ ")
print("3. Tentukan tingkat signifikan  $\alpha$ ")
sig = 0.05
print("     $\alpha =$ ", sig)
print("4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis")
daerahKritis = stats.norm.ppf(1-(sig/2))
print("    Didapatkan daerah kritis  $z >$ " + str(daerahKritis) + " atau  $z < -$ " + str(dae
rahKritis))
print("    Digunakan uji statistik z")
print("5. Hitung uji nilai statistik dari data sample")
ztest ,pval = stests.ztest(df["SumbuUtama"], x2=None, value=116)
print("    p value = ", float(pval))
print("    z value = ", ztest)
print("6. Ambil keputusan apakah H0 ditolak atau tidak")
if (ztest > daerahKritis or ztest < -daerahKritis):
    print("    H0 ditolak karena nilai ztest berada pada daerah kritis")
else:
    print("    H0 diterima karena nilai ztest tidak berada pada daerah kritis")
if (pval < 0.05):
    print("    H0 ditolak karena p value lebih kecil dari nilai signifikan")
else:
    print("    H0 diterima karena p value lebih besar dari nilai signifikan")
print("Boxplot:")
%matplotlib inline
boxplot = df.boxplot(column=["SumbuUtama"])
```

SOAL 4B

Nilai Rata-rata Sumbu Utama tidak sama dengan 116?

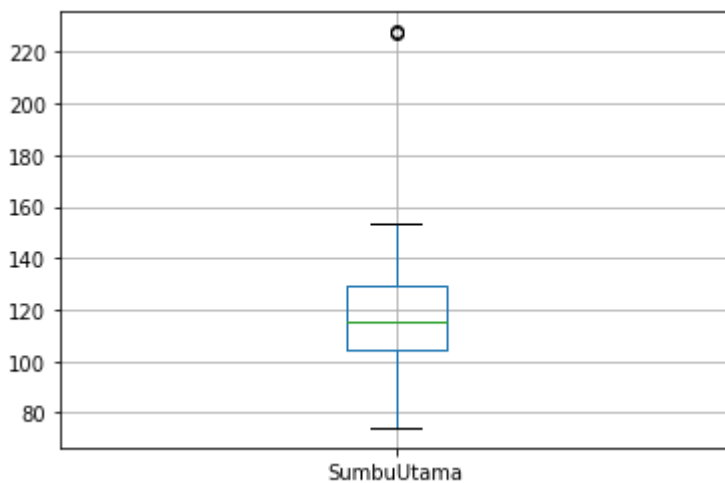
Enam langkah testing:

1. Tentukan hipotesis nol atau H_0
 $H_0: \mu = 116$
2. Tentukan hipotesis alternatif atau H_1
 $H_1: \mu \neq 116$
3. Tentukan tingkat signifikan α
 $\alpha = 0.05$
4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis
Didapatkan daerah kritis $z > 1.959963984540054$ atau $z < -1.959963984540054$

Digunakan uji statistik z

5. Hitung uji nilai statistik dari data sample
p value = 0.9559415872977383
z value = 0.05524712326731844
6. Ambil keputusan apakah H_0 ditolak atau tidak
 H_0 diterima karena nilai ztest tidak berada pada daerah kritis
 H_0 diterima karena p value lebih besar dari nilai signifikan

Boxplot:



In [248]:

```
print("SOAL 4C")
print("Nilai Rata-rata 20 baris pertama kolom Sumbu Kecil bukan 50?\n")
data = df.head(20)
print("Enam langkah testing:")
print("1. Tentukan hipotesis nol atau H0")
print("    H0:  $\mu = 50$ ")
print("2. Tentukan hipotesis alternatif atau H1")
print("    H1:  $\mu \neq 50$ ")
print("3. Tentukan tingkat signifikan  $\alpha$ ")
sig = 0.05
print("     $\alpha =$ ", sig)
print("4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis")
daerahKritis = stats.norm.ppf(1-(sig/2))
print("    Didapatkan daerah kritis  $z >$ " + str(daerahKritis) + " atau  $z < -$ " + str(dae
rahKritis))
print("    Digunakan uji statistik  $z$ ")
print("5. Hitung uji nilai statistik dari data sample")
ztest ,pval = stests.ztest(data["SumbuKecil"], x2=None, value=50)
print("    p value = ", float(pval))
print("    z value = ", ztest)
print("6. Ambil keputusan apakah H0 ditolak atau tidak")
if (ztest > daerahKritis or ztest < -daerahKritis):
    print("    H0 ditolak karena nilai ztest berada pada daerah kritis")
else:
    print("    H0 diterima karena nilai ztest tidak berada pada daerah kritis")
if (pval < 0.05):
    print("    H0 ditolak karena p value lebih kecil dari nilai signifikan")
else:
    print("    H0 diterima karena p value lebih besar dari nilai signifikan")
print("Boxplot:")
%matplotlib inline
boxplot = data.boxplot(column=["SumbuKecil"])
```

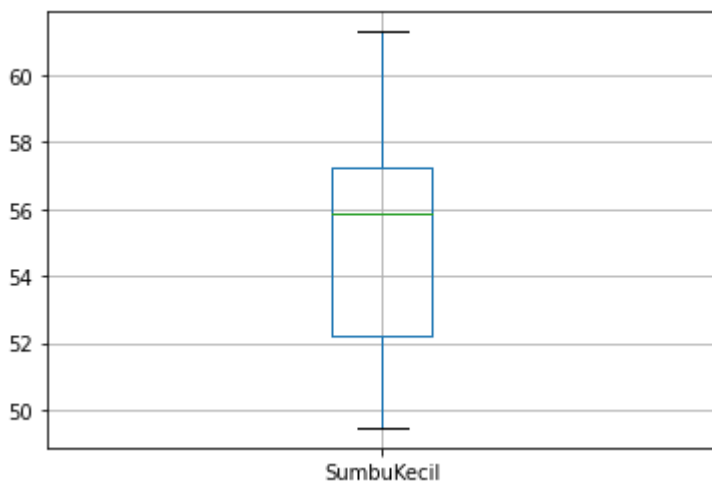
SOAL 4C

Nilai Rata-rata 20 baris pertama kolom Sumbu Kecil bukan 50?

Enam langkah testing:

1. Tentukan hipotesis nol atau H_0
 $H_0: \mu = 50$
2. Tentukan hipotesis alternatif atau H_1
 $H_1: \mu \neq 50$
3. Tentukan tingkat signifikan α
 $\alpha = 0.05$
4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis
Didapatkan daerah kritis $z > 1.959963984540054$ atau $z < -1.959963984540054$
Digunakan uji statistik z
5. Hitung uji nilai statistik dari data sample
 $p \text{ value} = 9.284238477468992e-11$
 $z \text{ value} = 6.478168916968894$
6. Ambil keputusan apakah H_0 ditolak atau tidak
 H_0 ditolak karena nilai z test berada pada daerah kritis
 H_0 ditolak karena p value lebih kecil dari nilai signifikan

Boxplot:



In [249]:

```
print("SOAL 4D")
print("Proporsi nilai Diameter yang lebih dari 85, adalah tidak sama dengan 15%?
\n")
print("Enam langkah testing:")
print("1. Tentukan hipotesis nol atau H0")
print("    H0 =  $\mu$  = 0.15")
print("2. Tentukan hipotesis alternatif atau H1")
print("    H1 =  $\mu \neq 0.15$  (two-tailed test)")
print("3. Tentukan tingkat signifikan  $\alpha$ ")
sig = 0.05
print("     $\alpha$  = ", sig)
print("4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis")
daerahKritis = norm.ppf(1-(sig/2))
print("    Didapatkan daerah kritis z > "+str(daerahKritis)+" atau z < -"+str(dae
rahKritis))
print("    Digunakan uji statistik z")
print("5. Hitung uji nilai statistik dari data sample")
df2 = df[df["Diameter"]>85]
ztest, pval = proportions_ztest(df2["Diameter"].count(),
                                df["Diameter"].count(),
                                0.15, alternative="two-sided")

print("    P-value =", pval)
print("    z =", ztest)
print("6. Ambil keputusan apakah H0 ditolak atau tidak")
if (ztest > daerahKritis or ztest < -daerahKritis):
    print("    H0 ditolak karena nilai ztest berada pada daerah kritis")
else:
    print("    H0 diterima karena nilai ztest tidak berada pada daerah kritis")
if (pval < 0.05):
    print("    H0 ditolak karena p value lebih kecil dari nilai signifikan")
else:
    print("    H0 diterima karena p value lebih besar dari nilai signifikan")
print("Boxplot:")
%matplotlib inline
boxplot = df.boxplot(column=["Diameter"])
```

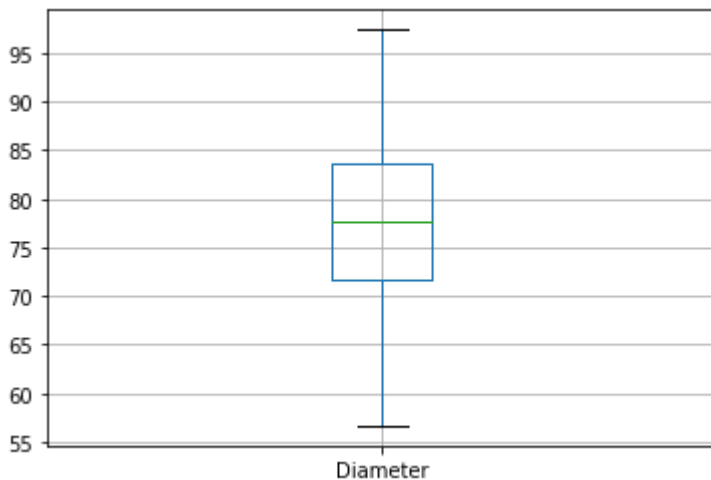
SOAL 4D

Proporsi nilai Diameter yang lebih dari 85, adalah tidak sama dengan 15%?

Enam langkah testing:

1. Tentukan hipotesis nol atau H_0
 $H_0 = \mu = 0.15$
2. Tentukan hipotesis alternatif atau H_1
 $H_1 = \mu \neq 0.15$ (two-tailed test)
3. Tentukan tingkat signifikan α
 $\alpha = 0.05$
4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis
Didapatkan daerah kritis $z > 1.959963984540054$ atau $z < -1.959963984540054$
Digunakan uji statistik z
5. Hitung uji nilai statistik dari data sample
 $P\text{-value} = 0.012842459209356831$
 $z = 2.4881083733392226$
6. Ambil keputusan apakah H_0 ditolak atau tidak
 H_0 ditolak karena nilai z test berada pada daerah kritis
 H_0 ditolak karena p value lebih kecil dari nilai signifikan

Boxplot:



In [255]:

```
print("SOAL 4E")
print("Proporsi nilai Keliling yang kurang dari 100, adalah kurang dari 5%?\n")
print("Enam langkah testing:")
print("1. Tentukan hipotesis nol atau H0")
print("    H0 =  $\mu$  = 0.05")
print("2. Tentukan hipotesis alternatif atau H1")
print("    H1 =  $\mu$  < 0.05 (one-tailed test)")
print("3. Tentukan tingkat signifikan  $\alpha$ ")
sig = 0.05
print("     $\alpha$  = ", sig)
print("4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis")
daerahKritis = norm.ppf(sig)
print("    Didapatkan daerah kritis z < ", -daerahKritis)
print("5. Hitung uji nilai statistik dari data sample")
df2 = df[df["Keliling"]<100]
print("    Tidak ada Keliling yang kurang dari 100")
ztest, pval = proportions_ztest(df2["Keliling"].count(),
                                df["Keliling"].count(),
                                0.05, alternative="smaller")

print("    P-value =", pval)
print("    z =", ztest)
print("6. Ambil keputusan apakah H0 ditolak atau tidak")
if (ztest < -daerahKritis):
    print("    H0 ditolak karena nilai ztest berada pada daerah kritis")
else:
    print("    H0 diterima karena nilai ztest tidak berada pada daerah kritis")
if (pval < 0.05):
    print("    H0 ditolak karena p value lebih kecil dari nilai signifikan")
else:
    print("    H0 diterima karena p value lebih besar dari nilai signifikan")
print("Boxplot:")
%matplotlib inline
boxplot = df.boxplot(column=["Keliling"])
```


SOAL 4E

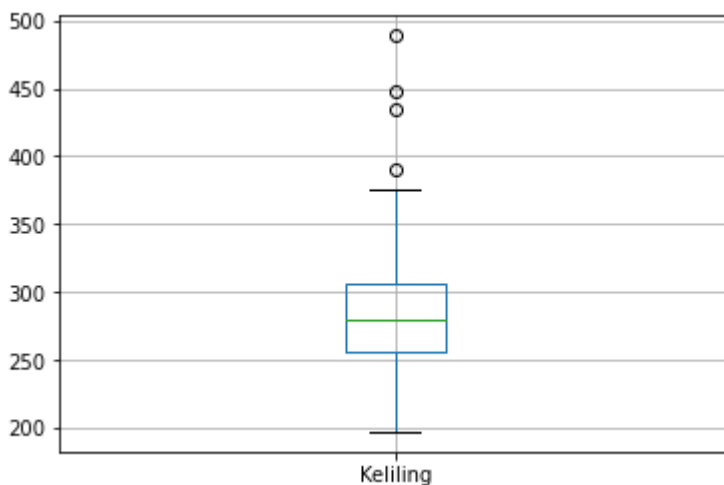
Proporsi nilai Keliling yang kurang dari 100, adalah kurang dari 5%?

Enam langkah testing:

1. Tentukan hipotesis nol atau H_0
 $H_0 = \mu = 0.05$
2. Tentukan hipotesis alternatif atau H_1
 $H_1 = \mu < 0.05$ (one-tailed test)
3. Tentukan tingkat signifikan α
 $\alpha = 0.05$
4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis
Didapatkan daerah kritis $z < 1.6448536269514729$
5. Hitung uji nilai statistik dari data sample
Tidak ada Keliling yang kurang dari 100
P-value = 0.0
 $z = -\text{inf}$
6. Ambil keputusan apakah H_0 ditolak atau tidak
 H_0 ditolak karena nilai ztest berada pada daerah kritis
 H_0 ditolak karena p value lebih kecil dari nilai signifikan

Boxplot:

```
/Users/gea/opt/anaconda3/lib/python3.8/site-packages/statsmodels/stats/weightstats.py:671: RuntimeWarning: divide by zero encountered in double_scalars
  zstat = value / std_diff
```



In [256]:

```
print("SOAL 5A")
print("Data kolom AreaBulatan dibagi 2 sama rata: bagian awal dan bagian akhir k
olom.")
print("Benarkah rata-rata kedua bagian tersebut sama?\n")
splitData = np.array_split(df,2)
sampleAwal = splitData[0].sample(100)
sampleAkhir = splitData[1].sample(100)
print("Enam langkah testing:")
print("1. Tentukan hipotesis nol atau H0")
print("    H0:  $\mu_1 - \mu_2 = 0$ ")
print("2. Tentukan hipotesis alternatif atau H1")
print("    H1:  $\mu_1 - \mu_2 \neq 0$ ")
print("3. Tentukan tingkat signifikan  $\alpha$ ")
sig = 0.05
print("     $\alpha =$ ", sig)
print("4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis")
daerahKritis = stats.norm.ppf(1-(sig/2))
print("    Digunakan uji statistik z-test")
print("    Daerah kritis =", daerahKritis)
print("5. Hitung uji nilai statistik dari data sample")
meanAwal = sampleAwal["AreaBulatan"].mean()
meanAkhir = sampleAkhir["AreaBulatan"].mean()
stdAwal = sampleAwal["AreaBulatan"].std()
stdAkhir = sampleAkhir["AreaBulatan"].std()
ztest = (meanAwal-meanAkhir)/math.sqrt(((stdAwal**2)/100)+((stdAkhir**2)/100))
print("    Nilai uji statistik (z) =", ztest)
pval = stats.norm.sf(abs(ztest))*2
print("    Nilai p =", pval)
print("6. Ambil keputusan apakah H0 ditolak atau tidak")
if (ztest > daerahKritis or ztest < -daerahKritis):
    print("    H0 ditolak karena nilai ztest berada di daerah kritis")
else:
    print("    H0 diterima karean nilai ztest berada di luar daerah kritis")
if (sig < pval):
    print("    H0 diterima karena nilai p lebih besar dari nilai signifikan")
else:
    print("    H0 ditolak karena nilai p lebih kecil dari nilai signifikan")
print("Boxplot:")
%matplotlib inline
boxplot1 = splitData[0].boxplot(column=["AreaBulatan"], color="blue")
boxplot2 = splitData[1].boxplot(column=["AreaBulatan"], color="red")
```

SOAL 5A

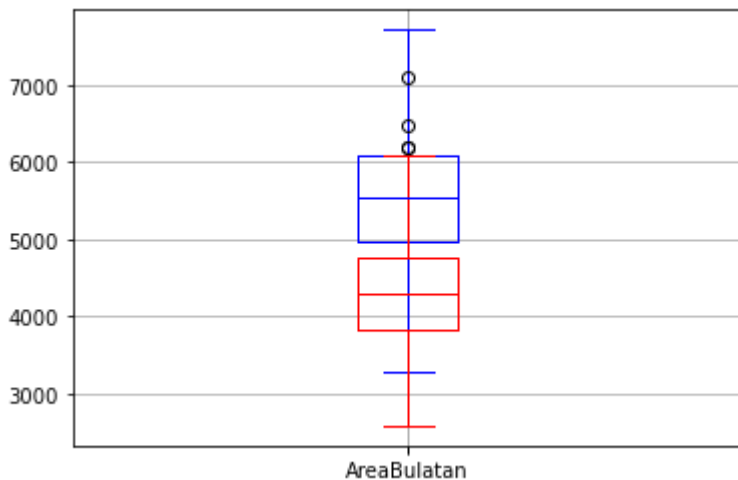
Data kolom AreaBulatan dibagi 2 sama rata: bagian awal dan bagian akhir kolom.

Benarkah rata-rata kedua bagian tersebut sama?

Enam langkah testing:

1. Tentukan hipotesis nol atau H_0
 $H_0: \mu_1 - \mu_2 = 0$
2. Tentukan hipotesis alternatif atau H_1
 $H_1: \mu_1 - \mu_2 \neq 0$
3. Tentukan tingkat signifikan α
 $\alpha = 0.05$
4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis
Digunakan uji statistik z-test
Daerah kritis = 1.959963984540054
5. Hitung uji nilai statistik dari data sample
Nilai uji statistik (z) = 11.344526317172132
Nilai p = 7.895261199362185e-30
6. Ambil keputusan apakah H_0 ditolak atau tidak
 H_0 ditolak karena nilai ztest berada di daerah kritis
 H_0 ditolak karena nilai p lebih kecil dari nilai signifikan

Boxplot:



In [257]:

```
print("SOAL 5B")
print("Data kolom Kadar Air dibagi 2 sama rata: bagian awal dan bagian akhir kolom.")
print("Benarkah rata-rata bagian awal lebih besar dari pada bagian akhir sebesar 0.2?\n")
splitData = np.array_split(df,2)
sampleAwal = splitData[0].sample(100)
sampleAkhir = splitData[1].sample(100)
print("Enam langkah testing:")
print("1. Tentukan hipotesis nol atau H0")
print("    H0:  $\mu_1 - \mu_2 = 0.2$ ")
print("2. Tentukan hipotesis alternatif atau H1")
print("    H1:  $\mu_1 - \mu_2 \neq 0.2$ ")
print("3. Tentukan tingkat signifikan  $\alpha$ ")
sig = 0.05
print("     $\alpha =$ ", sig)
print("4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis")
daerahKritis = stats.norm.ppf(1-(sig/2))
print("    Digunakan uji statistik z-test")
print("    Daerah kritis =", daerahKritis)
print("5. Hitung uji nilai statistik dari data sample")
meanAwal = sampleAwal["KadarAir"].mean()
meanAkhir = sampleAkhir["KadarAir"].mean()
stdAwal = sampleAwal["KadarAir"].std()
stdAkhir = sampleAkhir["KadarAir"].std()
ztest = (meanAwal-meanAkhir)-0.2/math.sqrt(((stdAwal**2)/100)+((stdAkhir**2)/100))
print("    Nilai uji statistik (z) =", ztest)
pval = stats.norm.sf(abs(ztest))*2
print("    Nilai p =", pval)
print("6. Ambil keputusan apakah H0 ditolak atau tidak")
if (ztest > daerahKritis or ztest < -daerahKritis):
    print("    H0 ditolak karena nilai ztest berada di daerah kritis")
else:
    print("    H0 diterima karena nilai ztest berada di luar daerah kritis")
if (sig < pval):
    print("    H0 diterima karena nilai p lebih besar dari nilai signifikan")
else:
    print("    H0 ditolak karena nilai p lebih kecil dari nilai signifikan")
print("Boxplot:")
%matplotlib inline
boxplot1 = splitData[0].boxplot(column=["KadarAir"], color="blue")
boxplot2 = splitData[1].boxplot(column=["KadarAir"], color="red")
```

SOAL 5B

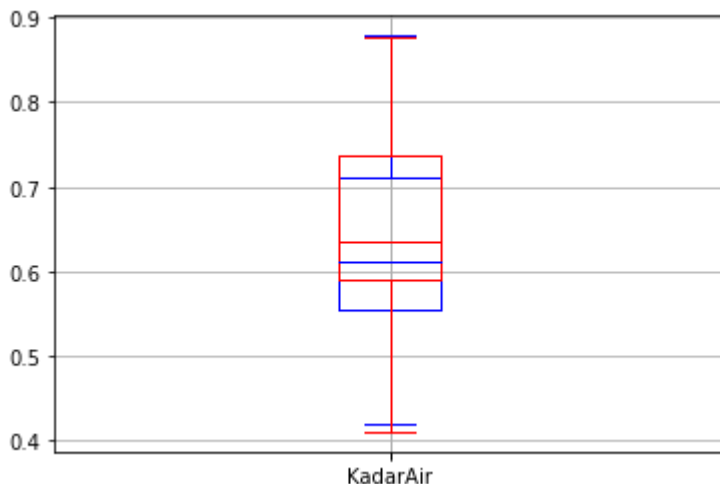
Data kolom Kadar Air dibagi 2 sama rata: bagian awal dan bagian akhir kolom.

Benarkah rata-rata bagian awal lebih besar dari pada bagian akhir sebesar 0.2?

Enam langkah testing:

1. Tentukan hipotesis nol atau H_0
 $H_0: \mu_1 - \mu_2 = 0.2$
2. Tentukan hipotesis alternatif atau H_1
 $H_1: \mu_1 - \mu_2 \neq 0.2$
3. Tentukan tingkat signifikan α
 $\alpha = 0.05$
4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis
Digunakan uji statistik z-test
Daerah kritis = 1.959963984540054
5. Hitung uji nilai statistik dari data sample
Nilai uji statistik (z) = -14.545437261622999
Nilai p = 6.243064754721968e-48
6. Ambil keputusan apakah H_0 ditolak atau tidak
 H_0 ditolak karena nilai ztest berada di daerah kritis
 H_0 ditolak karena nilai p lebih kecil dari nilai signifikan

Boxplot:



In [258]:

```
print("SOAL 5C")
print("Rata-rata 20 baris pertama kolom Bulatan sama dengan 20 baris terakhirnya?\n")
dataAwal = df.head(20)
dataAkhir = df.tail(20)
sampleAwal = dataAwal.sample(15)
sampleAkhir = dataAkhir.sample(15)
print("Enam langkah testing:")
print("1. Tentukan hipotesis nol atau H0")
print("    H0:  $\mu_1 - \mu_2 = 0$ ")
print("2. Tentukan hipotesis alternatif atau H1")
print("    H1:  $\mu_1 - \mu_2 \neq 0$ ")
print("3. Tentukan tingkat signifikan  $\alpha$ ")
sig = 0.05
print("     $\alpha =$ ", sig)
print("4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis")
daerahKritis = stats.norm.ppf(1-(sig/2))
print("    Digunakan uji statistik z-test")
print("    Daerah kritis =", daerahKritis)
print("5. Hitung uji nilai statistik dari data sample")
meanAwal = sampleAwal["Bulatan"].mean()
meanAkhir = sampleAkhir["Bulatan"].mean()
stdAwal = sampleAwal["Bulatan"].std()
stdAkhir = sampleAkhir["Bulatan"].std()
ztest = (meanAwal-meanAkhir)/math.sqrt(((stdAwal**2)/100)+((stdAkhir**2)/100))
print("    Nilai uji statistik (z) =", ztest)
pval = stats.norm.sf(abs(ztest))*2
print("    Nilai p =", pval)
print("6. Ambil keputusan apakah H0 ditolak atau tidak")
if (ztest > daerahKritis or ztest < -daerahKritis):
    print("    H0 ditolak karena nilai ztest berada di daerah kritis")
else:
    print("    H0 diterima karena nilai ztest berada di luar daerah kritis")
if (sig < pval):
    print("    H0 diterima karena nilai p lebih besar dari nilai signifikan")
else:
    print("    H0 ditolak karena nilai p lebih kecil dari nilai signifikan")
print("Boxplot:")
%matplotlib inline
boxplot1 = dataAwal.boxplot(column=["Bulatan"], color="blue")
boxplot2 = dataAkhir.boxplot(column=["Bulatan"], color="red")
```

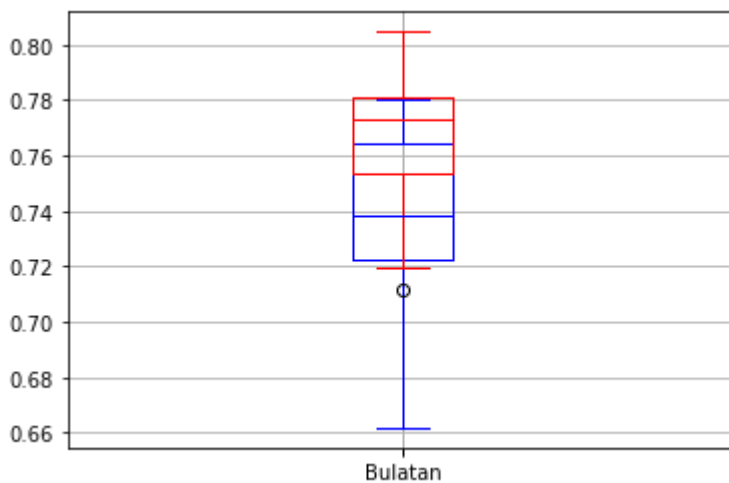
SOAL 5C

Rata-rata 20 baris pertama kolom Bulatan sama dengan 20 baris terakhirnya?

Enam langkah testing:

1. Tentukan hipotesis nol atau H_0
 $H_0: \mu_1 - \mu_2 = 0$
2. Tentukan hipotesis alternatif atau H_1
 $H_1: \mu_1 - \mu_2 \neq 0$
3. Tentukan tingkat signifikan α
 $\alpha = 0.05$
4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis
Digunakan uji statistik z-test
Daerah kritis = 1.959963984540054
5. Hitung uji nilai statistik dari data sample
Nilai uji statistik (z) = -7.820737283023162
Nilai p = 5.251484207670474e-15
6. Ambil keputusan apakah H_0 ditolak atau tidak
 H_0 ditolak karena nilai ztest berada di daerah kritis
 H_0 ditolak karena nilai p lebih kecil dari nilai signifikan

Boxplot:



In [259]:

```

print("SOAL 5D")
print("Proporsi nilai bagian awal Ransum yang lebih dari 2, adalah lebih besar d
aripada, proporsi nilai yang sama di bagian akhir Ransum?\n")
splitData = np.array_split(df,2)
sampleAwal = splitData[0].sample(100)
sampleAkhir = splitData[1].sample(100)
countAwal = 0
for i in sampleAwal["Ransum"]:
    if (i > 2):
        countAwal += 1
countAkhir = 0
for i in sampleAkhir["Ransum"]:
    if (i > 2):
        countAkhir += 1
p1 = countAwal/100
p2 = countAkhir/100
print("Enam langkah testing:")
print("1. Tentukan hipotesis nol atau H0")
print("    H0: p1 = p2")
print("2. Tentukan hipotesis alternatif atau H1")
print("    H1: p1 > p2")
print("3. Tentukan tingkat signifikan  $\alpha$ ")
sig = 0.05
print("     $\alpha$  =", sig)
print("4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis")
daerahKritis = stats.norm.ppf(1-sig)
print("    Digunakan uji statistik z-test")
print("    Daerah kritis =", daerahKritis)
print("5. Hitung uji nilai statistik dari data sample")
p = (countAwal+countAkhir)/200
ztest = (p1-p2)/math.sqrt((p*(1-p))*(1/100+1/100))
print("    Nilai uji statistik (z) =", ztest)
pval = stats.norm.sf(abs(ztest))
print("    Nilai p =", pval)
print("6. Ambil keputusan apakah H0 ditolak atau tidak")
if (ztest > daerahKritis):
    print("    H0 ditolak karena nilai ztest berada di daerah kritis")
else:
    print("    H0 diterima karean nilai ztest berada di luar daerah kritis")
if (sig < pval):
    print("    H0 diterima karena nilai p lebih besar dari nilai signifikan")
else:
    print("    H0 ditolak karena nilai p lebih kecil dari nilai signifikan")
print("Boxplot:")
%matplotlib inline
boxplot1 = splitData[0].boxplot(column=["Ransum"], color="blue")
boxplot2 = splitData[1].boxplot(column=["Ransum"], color="red")

```

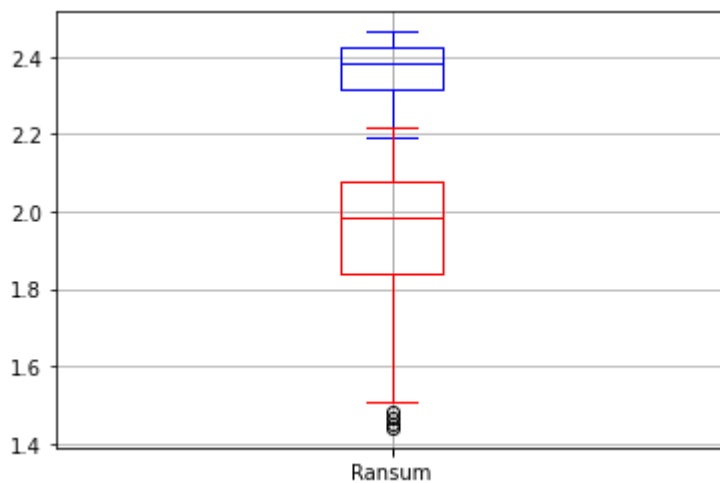

SOAL 5D

Proporsi nilai bagian awal Ransum yang lebih dari 2, adalah lebih besar daripada, proporsi nilai yang sama di bagian akhir Ransum?

Enam langkah testing:

1. Tentukan hipotesis nol atau H_0
 $H_0: p_1 = p_2$
2. Tentukan hipotesis alternatif atau H_1
 $H_1: p_1 > p_2$
3. Tentukan tingkat signifikan α
 $\alpha = 0.05$
4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis
Digunakan uji statistik z-test
Daerah kritis = 1.6448536269514722
5. Hitung uji nilai statistik dari data sample
Nilai uji statistik (z) = 8.382736442849094
Nilai p = 2.5855588862509366e-17
6. Ambil keputusan apakah H_0 ditolak atau tidak
 H_0 ditolak karena nilai ztest berada di daerah kritis
 H_0 ditolak karena nilai p lebih kecil dari nilai signifikan

Boxplot:



In [260]:

```
print("SOAL 5E")
print("Bagian awal kolom Diameter memiliki variansi yang sama dengan bagian akhir\nnya?")
print("")
splitData = np.array_split(df,2)
sampleAwal = splitData[0].sample(100)
sampleAkhir = splitData[1].sample(100)
print("Enam langkah testing:")
print("1. Tentukan hipotesis nol atau H0")
print("    H0:  $\sigma_1^2 = \sigma_2^2$ ")
print("2. Tentukan hipotesis alternatif atau H1")
print("    H1:  $\sigma_1^2 \neq \sigma_2^2$ ")
print("3. Tentukan tingkat signifikan  $\alpha$ ")
sig = 0.05
print("     $\alpha =$ ", sig)
print("4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis")
daerahKritis1 = stats.f.ppf(1-sig/2,99,99)
daerahKritis2 = 1-stats.f.ppf(sig/2,99,99)
print("    Digunakan uji statistik z-test")
print("    Daerah kritis 1 = ", daerahKritis1)
print("    Daerah kritis 2 = ", daerahKritis2)
print("5. Hitung uji nilai statistik dari data sample")
var1 = sampleAwal["Diameter"].std()**2
var2 = sampleAkhir["Diameter"].std()**2
ftest = var1/var2
print("    Nilai uji statistik (f) = ", ftest)
pval = stats.f.cdf(ftest,99,99)
print("    Nilai p = ", pval)
print("6. Ambil keputusan apakah H0 ditolak atau tidak")
if (ftest < daerahKritis2 or ftest > daerahKritis1):
    print("    H0 ditolak karena nilai ztest berada di daerah kritis")
else:
    print("    H0 diterima karean nilai ztest berada di luar daerah kritis")
if (sig < pval):
    print("    H0 diterima karena nilai p lebih besar dari nilai signifikan")
else:
    print("    H0 ditolak karena nilai p lebih kecil dari nilai signifikan")
print("Boxplot:")
%matplotlib inline
boxplot1 = splitData[0].boxplot(column=["Diameter"], color="blue")
boxplot2 = splitData[1].boxplot(column=["Diameter"], color="red")
```

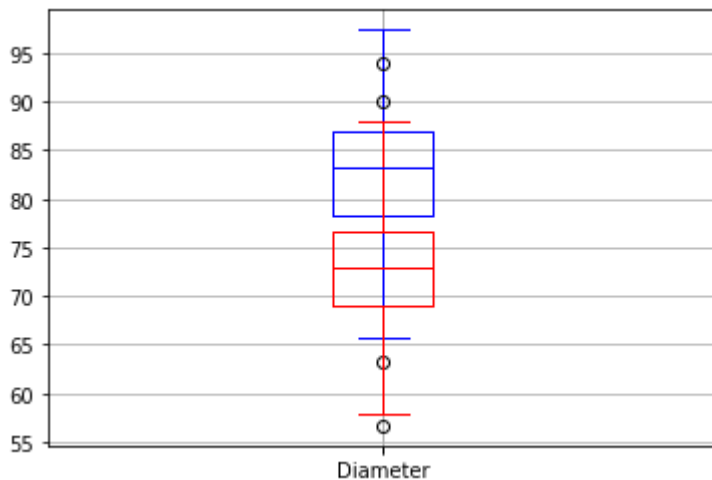
SOAL 5E

Bagian awal kolom Diameter memiliki variansi yang sama dengan bagian akhirnya?

Enam langkah testing:

1. Tentukan hipotesis nol atau H_0
 $H_0: \sigma_1^2 = \sigma_2^2$
2. Tentukan hipotesis alternatif atau H_1
 $H_1: \sigma_1^2 \neq \sigma_2^2$
3. Tentukan tingkat signifikan α
 $\alpha = 0.05$
4. Tentukan uji statistik yang sesuai dan tentukan daerah kritis
Digunakan uji statistik z-test
Daerah kritis 1 = 1.4862337676192938
Daerah kritis 2 = 0.3271583368733182
5. Hitung uji nilai statistik dari data sample
Nilai uji statistik (f) = 1.0772913351889826
Nilai p = 0.6440877197929458
6. Ambil keputusan apakah H_0 ditolak atau tidak
 H_0 diterima karean nilai ztest berada di luar daerah kritis
 H_0 diterima karena nilai p lebih besar dari nilai signifikan

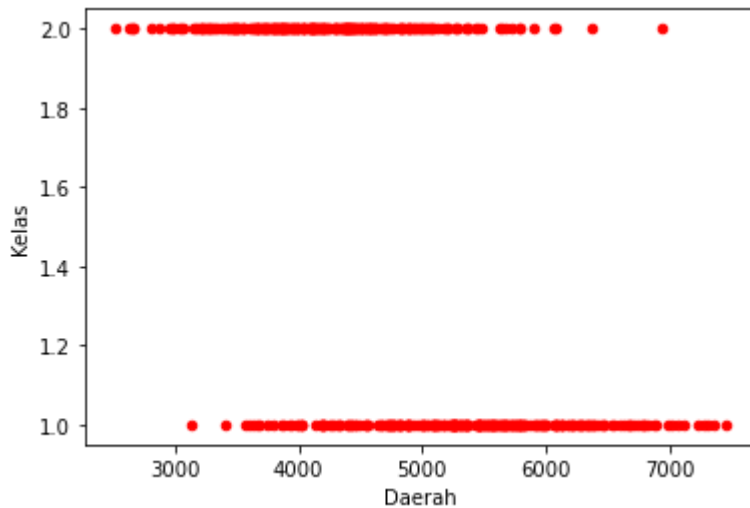
Boxplot:



In [261]:

```
print("SOAL 6: Scatter plot Kelas dengan Daerah")
df.plot(kind='scatter',x='Daerah',y='Kelas',color='red')
print("Correlation value =", df["Daerah"].corr(df["Kelas"]))
print("Daerah berkorelasi kuat negatif dengan kelas")
```

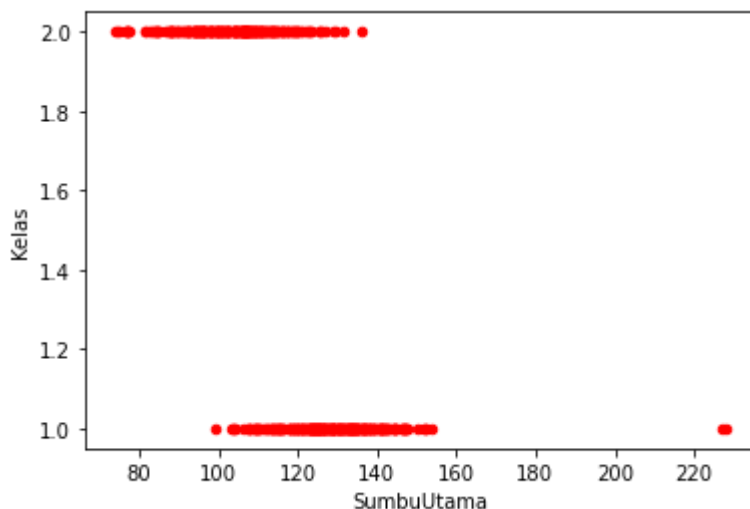
SOAL 6: Scatter plot Kelas dengan Daerah
Correlation value = -0.6027466517416654
Daerah berkorelasi kuat negatif dengan kelas



In [262]:

```
print("SOAL 6: Scatter plot Kelas dengan Sumbu Utama")
df.plot(kind='scatter',x='SumbuUtama',y='Kelas',color='red')
print("Correlation value =", df["SumbuUtama"].corr(df["Kelas"]))
print("Sumbu Utama berkorelasi kuat negatif dengan kelas")
```

SOAL 6: Scatter plot Kelas dengan Sumbu Utama
Correlation value = -0.7130906104204592
Sumbu Utama berkorelasi kuat negatif dengan kelas



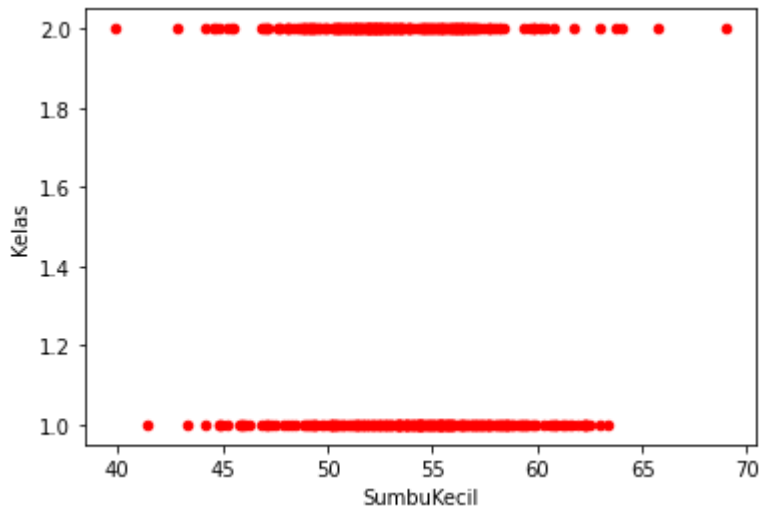
In [263]:

```
print("SOAL 6: Scatter plot Kelas dengan Sumbu Kecil")
df.plot(kind='scatter',x='SumbuKecil',y='Kelas',color='red')
print("Correlation value =", df["SumbuKecil"].corr(df["Kelas"]))
print("Sumbu Kecil berkorelasi sangat lemah negatif dengan kelas")
```

SOAL 6: Scatter plot Kelas dengan Sumbu Kecil

Correlation value = -0.15297517335535024

Sumbu Kecil berkorelasi sangat lemah negatif dengan kelas



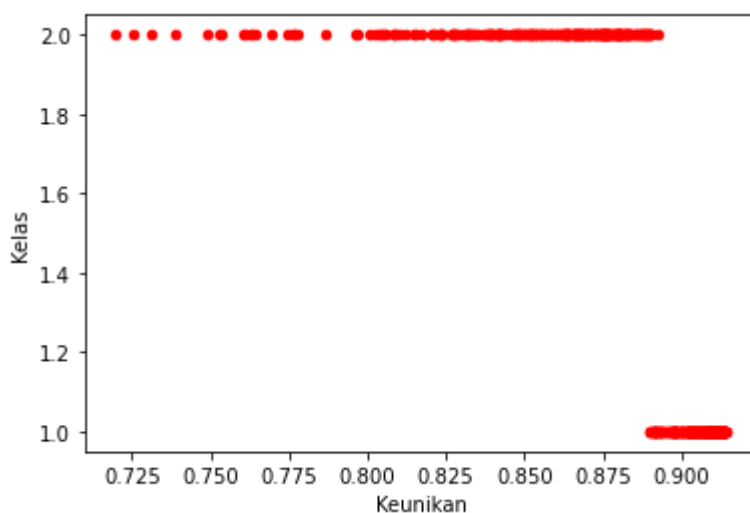
In [264]:

```
print("SOAL 6: Scatter plot Kelas dengan Keunikan")
df.plot(kind='scatter',x='Keunikan',y='Kelas',color='red')
print("Correlation value =", df["Keunikan"].corr(df["Kelas"]))
print("Keunikan berkorelasi kuat negatif dengan kelas")
```

SOAL 6: Scatter plot Kelas dengan Keunikan

Correlation value = -0.7304563686511927

Keunikan berkorelasi kuat negatif dengan kelas



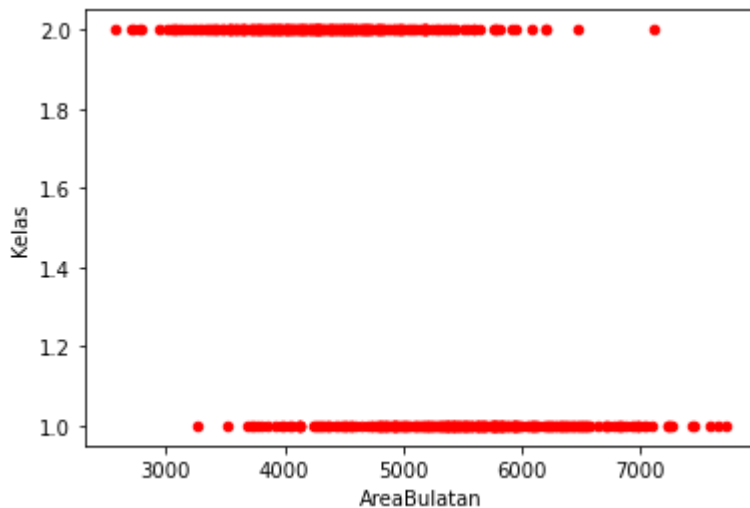
In [265]:

```
print("SOAL 6: Scatter plot Kelas dengan Area Bulatan")
df.plot(kind='scatter',x='AreaBulatan',y='Kelas',color='red')
print("Correlation value =", df["AreaBulatan"].corr(df["Kelas"]))
print("Area Bulatan berkorelasi kuat negatif dengan kelas")
```

SOAL 6: Scatter plot Kelas dengan Area Bulatan

Correlation value = -0.6073125434153747

Area Bulatan berkorelasi kuat negatif dengan kelas



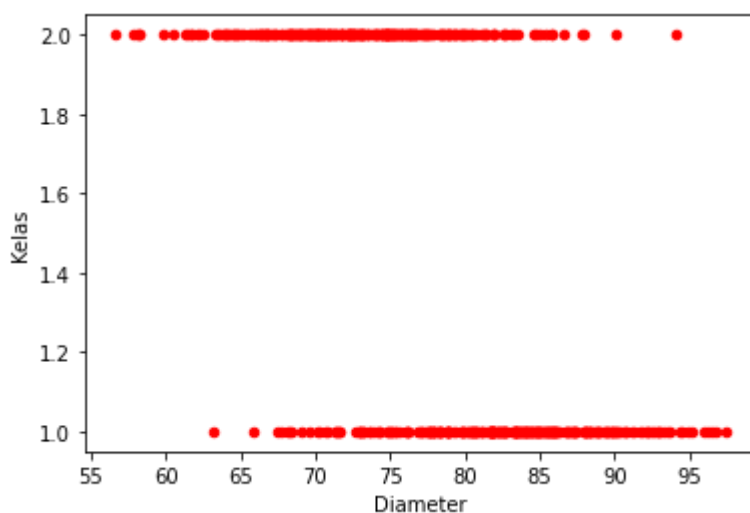
In [266]:

```
print("SOAL 6: Scatter plot Kelas dengan Diameter")
df.plot(kind='scatter',x='Diameter',y='Kelas',color='red')
print("Correlation value =", df["Diameter"].corr(df["Kelas"]))
print("Diameter berkorelasi kuat negatif dengan kelas")
```

SOAL 6: Scatter plot Kelas dengan Diameter

Correlation value = -0.6025356896618811

Diameter berkorelasi kuat negatif dengan kelas



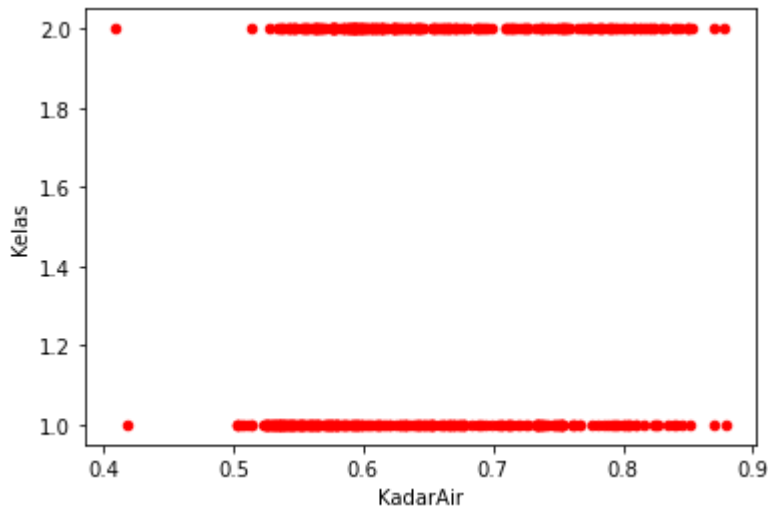
In [267]:

```
print("SOAL 6: Scatter plot Kelas dengan Kadar Air")
df.plot(kind='scatter',x='KadarAir',y='Kelas',color='red')
print("Correlation value =", df["KadarAir"].corr(df["Kelas"]))
print("Kadar Air berkorelasi sangat lemah positif dengan kelas")
```

SOAL 6: Scatter plot Kelas dengan Kadar Air

Correlation value = 0.13434422605727642

Kadar Air berkorelasi sangat lemah positif dengan kelas



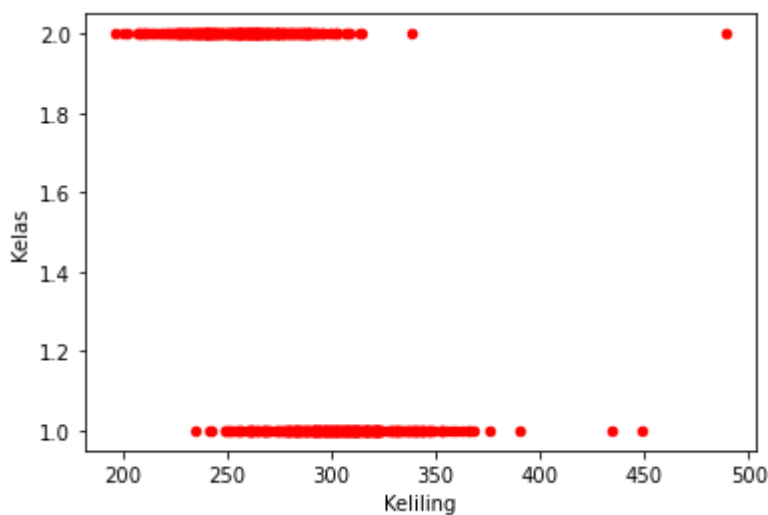
In [268]:

```
print("SOAL 6: Scatter plot Kelas dengan Keliling")
df.plot(kind='scatter',x='Keliling',y='Kelas',color='red')
print("Correlation value =", df["Keliling"].corr(df["Kelas"]))
print("Keliling berkorelasi kuat negatif dengan kelas")
```

SOAL 6: Scatter plot Kelas dengan Keliling

Correlation value = -0.6348607454756853

Keliling berkorelasi kuat negatif dengan kelas



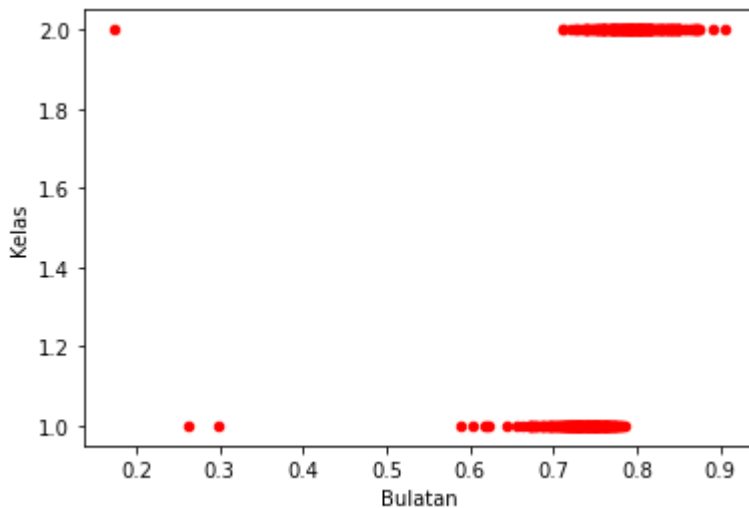
In [269]:

```
print("SOAL 6: Scatter plot Kelas dengan Bulatan")
df.plot(kind='scatter',x='Bulatan',y='Kelas',color='red')
print("Correlation value =", df["Bulatan"].corr(df["Kelas"]))
print("Bulatan berkorelasi sedang positif dengan kelas")
```

SOAL 6: Scatter plot Kelas dengan Bulatan

Correlation value = 0.5450045317240073

Bulatan berkorelasi sedang positif dengan kelas



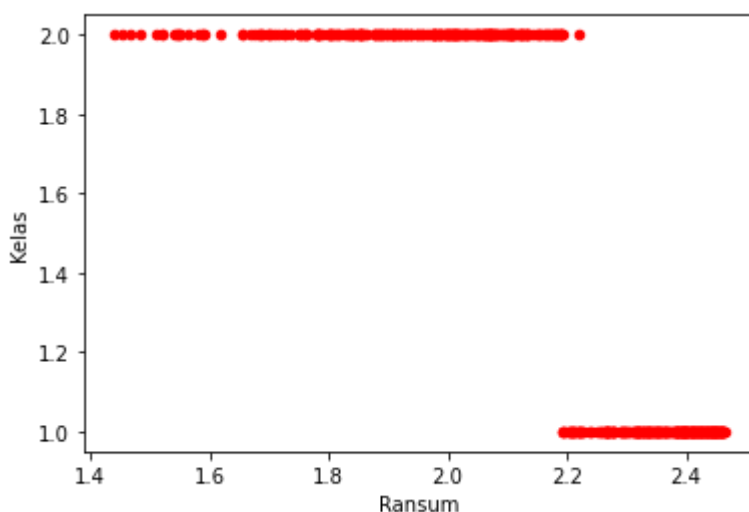
In [270]:

```
print("SOAL 6: Scatter plot Kelas dengan Ransum")
df.plot(kind='scatter',x='Ransum',y='Kelas',color='red')
print("Correlation value =", df["Ransum"].corr(df["Kelas"]))
print("Ransum berkorelasi sangat kuat negatif dengan kelas")
```

SOAL 6: Scatter plot Kelas dengan Ransum

Correlation value = -0.8399038681287486

Ransum berkorelasi sangat kuat negatif dengan kelas



In []: