In [1]:
```
from datascience import *
%matplotlib inline
import matplotlib.pyplot as plots
plots.style.use('fivethirtyeight')
import numpy as np
```

Lecture 25

Mean (= Average)

In [2]:
```
values = make_array(2, 3, 3, 9)
```

In [3]:
```
sum(values)/len(values)
```

Out[3]: 4.25

In [4]:
```
np.average(values)
```

Out[4]: 4.25

In [5]:
```
np.mean(values)
```

Out[5]: 4.25

In [6]:
```
(2 + 3 + 3 + 9)/4
```

Out[6]: 4.25

In [7]:
```
2*(1/4) + 3*(2/4) + 9*(1/4)
```

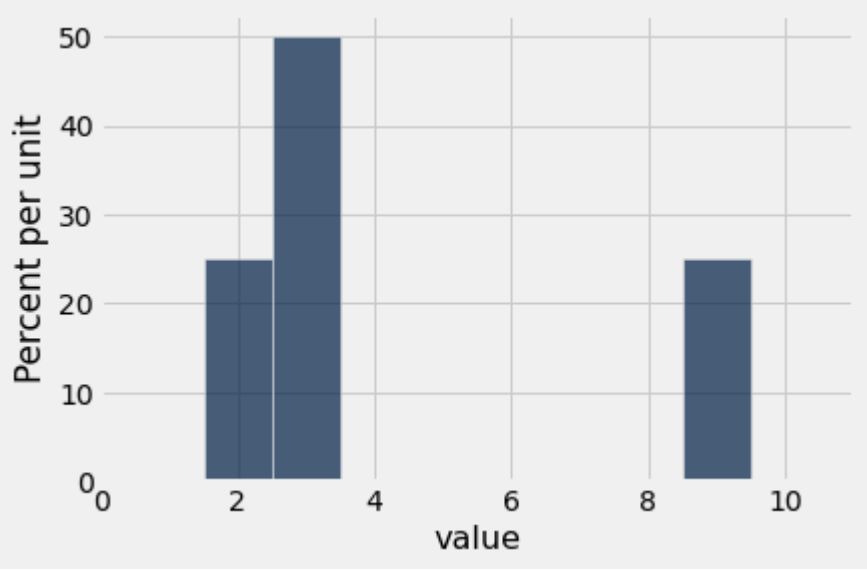
Out[7]: 4.25

In [8]:
```
values_table = Table().with_columns('value', values)
values_table
```

Out[8]:

value
2
3
3
9

In [9]:
```
bins_for_display = np.arange(0.5, 10.6, 1)
values_table.hist(0, bins = bins_for_display)
```



In [10]:
```
twos = 2 * np.ones(10)
threes = 3 * np.ones(20)
nines = 9 * np.ones(10)
```

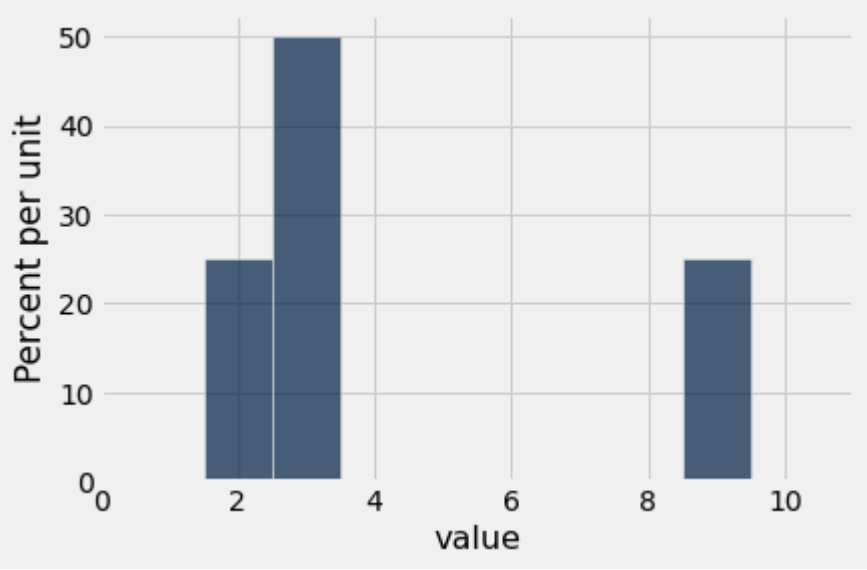
In [11]:
```
new_vals = np.append(np.append(twos, threes), nines)
```

In [12]:

```
len(new_vals)
```

Out[12]: 40

In [13]:
```
Table().with_column('value', new_vals).hist(bins = bins_for_display)
```



In [14]:
```
np.average(values)
```

Out[14]: 4.25

In [15]:
```
np.average(new_vals)
```

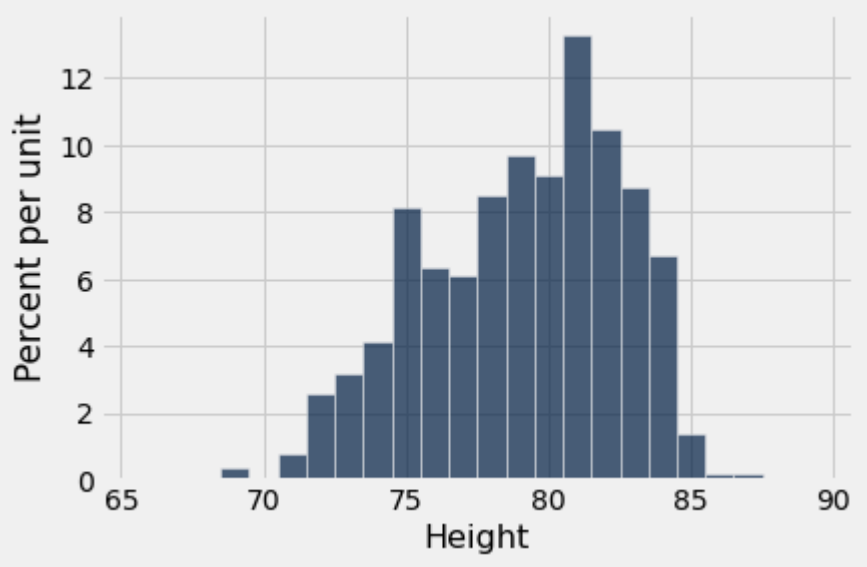
Out[15]: 4.25

In [16]:
```
nba = Table.read_table('nba2013.csv')
```

In [17]:
```
nba.labels
```

Out[17]: ('Name', 'Position', 'Height', 'Weight', 'Age in 2013')

In [18]:
```
nba.hist('Height', bins=np.arange(65.5, 90.5))
```



In [19]:
```
heights = nba.column('Height')
percentile(50, heights)
```

Out[19]:
```
80
```

In [20]:
```
np.mean(heights)
```

Out[20]:
```
79.06534653465347
```

Standard Deviation

In [21]:
```
sd_table = Table().with_columns('Value', values)
sd_table
```

Out[21]:

Value
2
3

Value
3
9

In [22]:
```
mean_value = np.mean(sd_table.column(0))
mean_value
```

Out[22]: 4.25

In [23]:
```
deviations = values - mean_value
sd_table = sd_table.with_column('Deviation', deviations)
sd_table
```

Out[23]:

Value	Deviation
2	-2.25
3	-1.25
3	-1.25
9	4.75

In [24]:
```
sum(deviations)
```

Out[24]: 0.0

In [25]:
```
sd_table = sd_table.with_columns('Squared Deviation', deviations ** 2)
sd_table
```

Out[25]:

Value	Deviation	Squared Deviation
2	-2.25	5.0625
3	-1.25	1.5625
3	-1.25	1.5625
9	4.75	22.5625

In [26]:
```
# Variance of the data

variance = np.mean(sd_table.column('Squared Deviation'))
variance
```

Out[26]: 7.6875

In [27]:
```
# Standard Deviation (SD) is the square root of the variance

sd = variance ** 0.5
sd
```

Out[27]: 2.7726341266023544

In [28]:
```
np.std(values)
```

Out[28]: 2.7726341266023544

In [ ]:
```

```