

In [20]: `df.head()`

Out[20]:

	Char Count	Value	Ratio
Word			
aa	2	2	1.000000
aah	3	10	3.333333
aahed	5	19	3.800000
aahing	6	40	6.666667
aahs	4	29	7.250000

▼ **What is the maximum value of Ratio ?**

In [21]: `df['Ratio'].max()`

Out[21]: 22.5

▼ **What word is the one with the highest Ratio ?**

In [22]: `df.sort_values(by='Ratio',ascending = False).head()`

Out[22]:

	Char Count	Value	Ratio
Word			
xu	2	45	22.500000
muzzy	5	111	22.200000
wry	3	66	22.000000
xyst	4	88	22.000000
pyx	3	65	21.666667

In [23]: `df.loc[df['Ratio'] == df['Ratio'].max()]`

Out[23]:

	Char Count	Value	Ratio
Word			
xu	2	45	22.5

▼ **How many words have a Ratio of 10 ?**

```
In [25]: df['Ratio'].value_counts()
```

```
Out[25]: Ratio
12.000000    3751
11.000000    3428
13.000000    3272
10.000000    2604
14.000000    2357
...
10.550000     1
8.944444     1
8.941176     1
9.263158     1
21.250000     1
Name: count, Length: 1333, dtype: int64
```

```
In [26]: df.loc[df['Ratio'] == 10].shape
```

```
Out[26]: (2604, 3)
```

```
In [27]: df.query('Ratio == 10').shape
```

```
Out[27]: (2604, 3)
```

▼ **What is the maximum Value of all the words with a Ratio of 10 ?**

```
In [29]: df.query('Ratio == 10').sort_values(by="Value", ascending = False).head()
```

```
Out[29]:
```

	Char Count	Value	Ratio
Word			
electrocardiographically	24	240	10.0
electroencephalographies	24	240	10.0
electroencephalographer	23	230	10.0
phonocardiographic	18	180	10.0
inconceivabilities	18	180	10.0

```
In [31]: df.loc[df['Ratio'] == 10, 'Value'].max()
```

```
Out[31]: 240
```

▼ **Of those words with a Value of 260, what is the lowest Char Count found?**

```
In [32]: df.query("Value == 260").sort_values(by="Char Count")
```

```
Out[32]:
```

	Char Count	Value	Ratio
Word			
hydroxytryptamine	17	260	15.294118
neuropsychologists	18	260	14.444444
psychophysiologist	18	260	14.444444
revolutionarinesses	19	260	13.684211
countermobilizations	20	260	13.000000
underrepresentations	20	260	13.000000

▼ *Find all the words where the char count is greater than the average*

```
In [33]: df['Char Count'].describe()
```

```
Out[33]: count    172821.000000
mean         9.087628
std          2.818285
min          2.000000
25%          7.000000
50%          9.000000
75%         11.000000
max         28.000000
Name: Char Count, dtype: float64
```

```
In [35]: mean_char_count = df['Char Count'].mean()
mean_char_count
```

```
Out[35]: 9.087628239623657
```

```
In [39]: df.query("`Char Count` > @mean_char_count")
```

```
Out[39]:
```

	Char Count	Value	Ratio
Word			
aardwolves	10	120	12.000000
abacterial	10	72	7.200000
abandoners	10	93	9.300000
abandoning	10	81	8.100000
abandonment	11	103	9.363636
...
zygomorphies	12	176	14.666667
zygomorphy	10	168	16.800000
zygosities	10	154	15.400000
zygospores	10	165	16.500000
zymologies	10	146	14.600000

67582 rows × 3 columns

```
In [43]: filtered_df = df[df['Char Count'] > mean_char_count]
filtered_df
```

```
Out[43]:
```

	Char Count	Value	Ratio
Word			
aardwolves	10	120	12.000000
abacterial	10	72	7.200000
abandoners	10	93	9.300000
abandoning	10	81	8.100000
abandonment	11	103	9.363636
...
zygomorphies	12	176	14.666667
zygomorphy	10	168	16.800000
zygosities	10	154	15.400000
zygospores	10	165	16.500000
zymologies	10	146	14.600000

67582 rows × 3 columns

```
In [44]: filtered_df = df[df['Char Count'] > mean_char_count]
```

▼ ***Based on the previous task, what word is it?***

```
In [ ]:
```