1. # Write your word_length_dictionary function here:

```
def word_length_dictionary(words):
word_lengths = {}
for word in words:
word_lengths[word] = len(word)
return word_lengths
# Uncomment these function calls to test your function:
print(word_length_dictionary(["apple", "dog", "cat"]))
# should print {"apple":5, "dog": 3, "cat":3}
print(word_length_dictionary(["a", ""]))
# should print {"a": 1, "": 0}
```

2. # Write your frequency_dictionary function here:

```
def frequency_dictionary(words):
freqs = {}
for word in words:
if word not in freqs:
freqs[word] = 0
freqs[word] += 1
return freqs
# Uncomment these function calls to test your function:
print(frequency_dictionary(["apple", "apple", "cat", 1]))
# should print {"apple":2, "cat":1, 1:1}
print(frequency_dictionary([0,0,0,0,0]))
# should print {0:5}
```

3. # Write your unique_values function here:

```
def unique_values(my_dictionary):
seen_values = []
for value in my_dictionary.values():
if value not in seen_values:
seen_values.append(value)
return len(seen_values)
# Uncomment these function calls to test your function:
print(unique_values({0:3, 1:1, 4:1, 5:3}))
# should print 2
```

```
print(unique_values({0:3, 1:3, 4:3, 5:3}))
# should print 1
```

4.
```
# Write your count_first_letter function here:
def count_first_letter(names):
letters = {}
for key in names:
first_letter = key[0]
if first_letter not in letters:
letters[first_letter] = 0
letters[first_letter] += len(names[key])
return letters
# Uncomment these function calls to test your function:
print(count_first_letter({"Stark": ["Ned", "Robb", "Sansa"], "Snow" : ["Jon"], "Lannister":
["Jaime", "Cersei", "Tywin"]}))
# should print {"S": 4, "L": 3}
print(count_first_letter({"Stark": ["Ned", "Robb", "Sansa"], "Snow" : ["Jon"], "Sannister":
["Jaime", "Cersei", "Tywin"]}))
# should print {"S": 7}
```