

## CSE 160 Section 5 Problems

---

**Dictionaries** are data types that map keys to values. Let's create some dictionaries:

```
my_dict = {} # empty dictionary
my_dict = dict() # empty dictionary
my_dict = { key_1: value_1, key_2: value_2, ... key_n: value_n } # initialized
```

Below are some useful dictionary operations, try them out to better understand what they do!

Dictionary Operation	Code
Access individual elements	<code>my_dict[key]</code>
Access all keys or all values	<code>my_dict.keys()</code> returns all the keys <code>my_dict.values()</code> returns all the values
Access all key value pairs	<code>my_dict.items()</code> returns a list of (key,value) pairs
Modifying the value of a key	<code>my_dict[key] = new_value</code>
Remove an item based on key	<code>del my_dict[key_i]</code>

- 
1. Create `def get_squares(number_list)` that accepts a list of numbers and returns a dictionary mapping each number in the list to its square.  
ex. `get_squares([1, 4, 4])` should return `{1:1, 4:16}`

```
def get_squares(number_list):
```

2. Write `def coldest_city(city_temperatures)` that takes in a dictionary and returns the city (key) with the lowest temperature (value).

Ex. With `city_temperatures = {'Seattle': 36, "Cupertino": 39, 'New York' : 57}`, `coldest_city(city_temperatures)` returns "Seattle".

```
def coldest_city(city_temperatures):
```

3. Write `def pokemon_types(pokemon_dict)` that takes parameter `pokemon_dict` and
  - a. returns a dict mapping each **type** of pokemon to its frequency in `pokemon_dict`

For example, when `pokemon_dict = {"pikachu": "electric", "charmander": "fire", "charizard" : "fire"}`,  
`pokemon_types(pokemon_dict)` returns {"electric" : 1, "fire" : 2}

```
def pokemon_types(pokemon_dict):
```

- b. returns a dict mapping each **type** of pokemon to a list of pokemon with that type.

```
For example, when pokemon_dict = {"pikachu": "electric",  
"charmander": "fire", "charizard": "fire"},  
pokemon_types(pokemon_dict) returns {'electric': ["pikachu"],  
'fire': ["charmander", "charizard"]}
```

```
def pokemon_types(pokemon_dict):
```

- 4. Write a function `def frequency(word)` that takes a String parameter, and returns a dictionary that maps each character in the String to its frequency. For example, `frequency("Star Wars")` returns `{"S": 1, "t": 1, "a": 2, "r": 2, " ": 1, "W": 1, "s": 1}`.