Python Exercises for Tuples, Dictionaries, and Sets

Tuples

1. Creating and Accessing Tuples

   - Create a tuple `my_tuple` containing the numbers `1`, `2`, and `3`. Print the tuple.

   - Access and print the second element of the tuple.

   - Try changing the first element of the tuple to `10` and observe what happens.

2. Tuple Packing and Unpacking

   - Create a tuple `person` containing your name, age, and city.

   - Unpack the tuple into three variables: `name`, `age`, and `city`.

   - Print each variable.

3. Nested Tuples

   - Create a tuple `nested_tuple` that contains two tuples: `(1, 2, 3)` and `(4, 5, 6)`.

   - Access and print the number `5` from `nested_tuple`.

4. Tuple Methods

   - Given the tuple `numbers = (1, 2, 3, 2, 4, 2)`, use the `count()` method to find how many times `2` appe

   - Use the `index()` method to find the position of the first occurrence of `3` in the tuple.

Dictionaries

1. Creating and Accessing Dictionaries

   - Create a dictionary `student` with keys `name`, `age`, and `grade`, and assign appropriate values.

   - Print the value associated with the key `name`.

   - Add a new key `school` with a value to the dictionary, then print the updated dictionary.

2. Modifying Dictionaries

   - Update the `age` of the `student` dictionary to a new value.

   - Remove the `grade` key from the dictionary.

   - Check if the key `grade` exists in the dictionary.

3. Iterating Over Dictionaries

   - Given the dictionary `capitals = {'France': 'Paris', 'Spain': 'Madrid', 'Japan': 'Tokyo'}`,

     write a loop that prints each country and its capital in the format: 'The capital of [country] is [capital].'

4. Dictionary Methods

   - Use the `keys()`, `values()`, and `items()` methods on the `capitals` dictionary and print the results.

   - Use the `get()` method to retrieve the capital of `Germany`, providing a default value of 'Not Found'.

5. Counting Characters

   - Write a program that counts the number of times each character appears in a given string. For example

     text = 'hello'

     Expected output: {'h': 1, 'e': 1, 'l': 2, 'o': 1}

Sets

1. Creating and Using Sets

   - Create a set `my_set` containing the numbers `1`, `2`, `3`, `4`, `5`.

   - Add the number `6` to the set.

   - Try adding the number `3` again and observe what happens.

   - Remove the number `2` from the set.

2. Set Operations

   - Given two sets:

     set_a = {1, 2, 3, 4}

     set_b = {3, 4, 5, 6}

     - Find the union of `set_a` and `set_b`.

     - Find the intersection of `set_a` and `set_b`.

     - Find the difference between `set_a` and `set_b` (elements in `set_a` but not in `set_b`).

     - Find the symmetric difference between `set_a` and `set_b`.

3. Unique Elements

   - Given a list with duplicate elements:

numbers = [1, 2, 2, 3, 4, 4, 5]

- Use a set to remove duplicates and print the list of unique numbers.

4. Membership Testing

- Check if the number `3` is in `set_a`.

- Check if the number `6` is not in `set_a`.

5. Set Methods

- Use the `add()` method to add an element to a set.

- Use the `remove()` method to remove an element from a set. What happens if the element does not exi

- Use the `discard()` method to remove an element. How is it different from `remove()`?