

Python Data Structures - Exercises

Lists

1. Create a list of 5 fruits. Print the second fruit and check if 'apple' exists in the list.

```
# task 1
fruits = ["apple", "banana", "orange", "grape", "pear"]
print("Second fruit:", fruits[1])
print("Is 'apple' in the list?", "apple" in fruits)
```

✓ 0.0s

Second fruit: banana
Is 'apple' in the list? True

2. Add 'mango' to the list, remove one fruit, and then sort the list in ascending order.

```
# task 2
fruits.append("mango")
fruits.remove("banana")
fruits.sort()
print("Updated list:", fruits)
```

✓ 0.0s

Updated list: ['apple', 'grape', 'mango', 'orange', 'pear']

3. Take two lists: list1 = [1, 2, 3], list2 = [4, 5, 6]. Join them and print the result.

```
# task 3
list1 = [1, 2, 3]
list2 = [4, 5, 6]
joined_list = list1 + list2
print("Joined list:", joined_list)
```

✓ 0.0s

Joined list: [1, 2, 3, 4, 5, 6]

Tuples

1. Create a tuple of 5 numbers. Print the first and last elements.

```
# task 1
numbers = (10, 20, 30, 40, 50)
print("First element:", numbers[0])
print("Last element:", numbers[-1])
```

✓ 0.0s

First element: 10
Last element: 50

2. Write a program to find the index of a number in a tuple.

```
# task 2
numbers = (10, 20, 30, 40, 50)
index_of_30 = numbers.index(30)
print("Index of 30:", index_of_30)
```

✓ 0.0s

Index of 30: 2

3. Convert a tuple into a list, add one new element, and convert it back to a tuple.

```
# task 3
numbers = (10, 20, 30, 40, 50)
num_list = list(numbers)
num_list.append(60)
numbers = tuple(num_list)
print("Updated tuple:", numbers)
```

✓ 0.0s

Updated tuple: (10, 20, 30, 40, 50, 60)

Dictionaries

1. Create a dictionary with keys 'name', 'age', and 'city'. Print the value of 'name'.

```
# task 1
person = {"name": "Ali", "age": 25, "city": "Gilgit"}
print("Name:", person["name"])
✓ 0.0s
Name: Ali
```

2. Add a new key 'country' with a value. Update 'age' to a new number.

```
# task 2
person = {"name": "Ali", "age": 25, "city": "Gilgit"}
person["country"] = "Pakistan"
person["age"] = 26
print("Updated dictionary:", person)
✓ 0.0s
Updated dictionary: {'name': 'Ali', 'age': 26, 'city': 'Gilgit', 'country': 'Pakistan'}
```

3. Write a program that loops through a dictionary and prints all keys and values.

```
# task 3
person = {"name": "Ali", "age": 25, "city": "Gilgit"}
for key, value in person.items():
    print(key, ":", value)
[14] ✓ 0.0s
... name : Ali
age : 25
city : Gilgit
```

Sets

1. Create a set of 5 numbers. Check if the number 3 exists in the set.

```
# task 1
numbers_set = {1, 2, 3, 4, 5}
print("Is 3 in the set?", 3 in numbers_set)
```

✓ 0.0s

Is 3 in the set? True

2. Add a new number to the set and remove an existing number.

```
# task 2
numbers_set = {1, 2, 3, 4, 5}
numbers_set.add(6)
numbers_set.remove(2)
print("Updated set:", numbers_set)
```

✓ 0.0s

Updated set: {1, 3, 4, 5, 6}

3. Create two sets: set1 = {1, 2, 3, 4}, set2 = {3, 4, 5, 6}. Find the union, intersection, and difference.

```
# task 3
set1 = {1, 2, 3, 4}
set2 = {3, 4, 5, 6}
print("Union:", set1.union(set2))
print("Intersection:", set1.intersection(set2))
print("Difference:", set1.difference(set2))
```

✓ 0.0s

Union: {1, 2, 3, 4, 5, 6}
Intersection: {3, 4}
Difference: {1, 2}