Chapter 3: Model Solutions

Below, you'll find sample solutions to the lab exercises in the book.

# Lab Exercises 3.1

1. Program to sum all elements in a given list:

my\_list = [1, 2, 3, 4, 5]

total = sum(my\_list)

print("Sum of elements:", total)

2. Program to count the number of even and odd numbers in a given list:

my\_list = [1, 2, 3, 4, 5]

count\_even = sum(1 for num in my\_list if num % 2 == 0)

count\_odd = len(my\_list) - count\_even

print("Even count:", count\_even)

print("Odd count:", count\_odd)

3. Program to check if a given list is sorted in ascending order:

my\_list = [1, 2, 3, 4, 5]

sorted\_check = my\_list == sorted(my\_list)

if sorted\_check:

print("The list is sorted in ascending order.")

else:

print("The list is not sorted in ascending order.")

4. Program to reverse a given list:

my\_list = [1, 2, 3, 4, 5]

reversed\_list = my\_list[::-1]

print("Reversed list:", reversed\_list)

5. Program to find the frequency of each element in a given list:

my\_list = [1, 2, 1, 3, 2, 4, 2, 5]

freq\_dict = {}

for element in my\_list:

freq\_dict[element] = freq\_dict.get(element, 0) + 1

print("Frequency of elements:", freq\_dict)

# Lab Exercises 3.2

1. Program to perform set operations on two sets:

set1 = {1, 2, 3}

set2 = {3, 4, 5}

union\_set = set1.union(set2)

intersection\_set = set1.intersection(set2)

difference\_set = set1.difference(set2)

print("Union:", union\_set)

print("Intersection:", intersection\_set)

print("Difference:", difference\_set)

2. Program to modify a tuple of integers:

my\_tuple = (1, 2, 3, 4, 5)

modified\_tuple = tuple(num \* 2 if num % 2 == 0 else num + 1 for num in my\_tuple)

print("Modified tuple:", modified\_tuple)

3. Program to print names in alphabetical order:

my\_tuple = ('Alice', 'Bob', 'Charlie', 'David')

sorted\_names = tuple(sorted(my\_tuple))

print("Names in alphabetical order:", sorted\_names)

4. Program to create a set of unique characters from a string:

my\_string = "hello world"

unique\_chars = set(my\_string)

print("Unique characters:", unique\_chars)

5. Program to perform set operations on two sets:

set1 = {1, 2, 3}

set2 = {3, 4, 5}

union\_set = set1.union(set2)

intersection\_set = set1.intersection(set2)

symmetric\_diff\_set = set1.symmetric\_difference(set2)

print("Union:", union\_set)

print("Intersection:", intersection\_set)

print("Symmetric Difference:", symmetric\_diff\_set)

6. A Python program that takes a tuple of numbers and calculates the sum and average

my\_tuple = (10, 20, 30, 40, 50)

# Calculate the sum of the numbers in the tuple

total = sum(my\_tuple)

# Calculate the average of the numbers in the tuple

average = total / len(my\_tuple)

print("Sum:", total)

print("Average:", average)

# Lab Exercises 3.3

1. Program to create a dictionary representing a person's contact information:

contact\_info = {}

contact\_info['name'] = input("Enter name: ")

contact\_info['email'] = input("Enter email: ")

contact\_info['phone'] = input("Enter phone number: ")

print("Contact Information:", contact\_info)

2. Program to simulate an English to Spanish dictionary:

dictionary = {

'hello': 'hola',

'goodbye': 'adiós',

'thank you': 'gracias',

'yes': 'sí',

'no': 'no'

}

word = input("Enter an English word: ")

if word in dictionary:

translation = dictionary[word]

print("Spanish translation:", translation)

else:

print("Word not found in dictionary.")

3. Track the sales of products in a store.

sales = {}

def add\_sale():

product = input("Enter the name of the product: ")

price = float(input("Enter the price of the product: "))

if product in sales:

sales[product] += price

else:

sales[product] = price

print("Sale added.")

def display\_sales():

print("Sales Information:")

for product, amount in sales.items():

print(product + ": $" + str(amount))

while True:

print("\n=== Sales Tracking System ===")

print("1. Add Sale")

print("2. Display Sales")

print("3. Exit")

choice = int(input("Enter your choice (1-3): "))

if choice == 1:

add\_sale()

elif choice == 2:

display\_sales()

elif choice == 3:

break

else:

print("Invalid choice. Please try again.")

4. To merge two dictionaries, you can use the update() method.

dict1 = {'a': 1, 'b': 2}

dict2 = {'c': 3, 'd': 4}

dict1.update(dict2)

print(dict1)

5. Define a dictionary

my\_dict = {'key1': 'value1',   
 'key2': 'value2',   
 'key3': 'value3'}

6. Access an item in a dictionary

my\_dict = {'name': 'John', 'age': 30, 'city': 'New York'}

print(my\_dict['name'])

print(my\_dict['age'])

print(my\_dict['city'])

7. Use the get() method

my\_dict = {'name': 'John', 'age': 30}

print(my\_dict.get('name'))

print(my\_dict.get('city', 'Unknown'))

8. Use the pop() method

my\_dict = {'name': 'John', 'age': 30}

age = my\_dict.pop('age')

print(age)

print(my\_dict)

9. use the update() method

dict1 = {'a': 1, 'b': 2}

dict2 = {'c': 3, 'd': 4}

dict1.update(dict2)

print(dict1)