Assignement 5
Python

Question 1:
Variables and Data Types

a. Declare three variables: name (string), age (integer), and height (float). Assign values to them and print their types.
b. Create a tuple with three elements: your favorite color, number, and a boolean indicating whether you like Python.
c. Create a tuple with three elements: your first name, last name, and age. Unpack the tuple and print each element separately.
d. Create different type of variables and print their types

Question 2:
Working with Lists

a. Create a list of your favorite fruits. Print the length of the list, add one more fruit to the list, and print the updated list.
b. Create a list of 5 numbers. Print the first three elements, and then print the last two elements.
c. Create a list of words. Sort the list alphabetically and print the sorted list.
d. Given a list of characters, extract a sublist containing the elements from index 2 to 5 (inclusive).
e. Create a list of your favorite colors. Print the list in reverse order using list slicing.
f. Generate a list of numbers from 1 to 15. Print the elements at odd indices using list slicing.
g. Create a list of letters. Print every second element using list slicing.
h. Given a list of fruits, create a new list that is a copy of the original list using list slicing.

Question 3:
Conditional Statements

a. Write a program that takes an integer as input and prints whether it is positive, negative, or zero.
b. Write a program that checks whether a given number is even or odd. Print an appropriate message.
c. Ask the user to input a year. Determine if it's a leap year and print the result. A leap year is divisible by 4, but not by 100 unless it is divisible by 400.
d. Create a simple menu with options 1, 2, and 3. Ask the user to enter their choice and print a corresponding message for each choice.
e. Prompt the user to enter three numbers. Determine and print the largest among them.

Question 4:
Loops

a. Write a program that prints the first 5 multiples of a number entered by the user.
b. Create a list of numbers. Use a loop to iterate through the list and print only the even numbers.
c. Use a for loop and the range function to print the square of numbers from 1 to 5.
d. Write a program that asks the user to guess a secret number between 1 and 10. Keep prompting until they guess correctly.
e. Write a program that asks the user for an integer and calculates its factorial using a for loop. Print the result.
f. Write a sequence of programs that prints all the patterns you have learned in C language (lecture 4)
g. Create a simple password checker that asks the user to enter a password. Use a while loop to repeatedly prompt until the correct password is entered.

Question 5:
Functions

a. Create a function called calculate_area that takes the radius of a circle as an argument and returns its area. Test the function with a radius of 5.
b. Write a function called calculate_discount that takes the original price and discount percentage as arguments and returns the discounted price.
c. Write a function called greet_user that takes a name as a parameter and prints a personalized greeting. If no name is provided, use "Guest" as the default.
d. Write a function called add_element that takes a list and an element as parameters and adds the element to the list. Test the function with a list of your choice.
e. Write a function called calculate_power that takes two parameters (base and exponent) and returns the result of base raised to the power of the exponent.

Question 6:
Strings

a. Given a string, reverse it and print the result.
b. Ask the user to enter their full name. Print the length of the name and convert it to uppercase.
c. Ask the user for their age and name. Print a message that includes both pieces of information in a formatted string.
d. Ask the user to enter their favorite color. Print a message that repeats their favorite color five times using string concatenation.
e. Ask the user to enter a sentence. Print the sentence in uppercase and lowercase.

Question 7:
File Handling

a. Write a program that reads a text file named "sample.txt" and prints its contents.
b. Write a program that opens a file named "output.txt" and writes the message "Hello, File!" to it.
c. Write a program that reads the contents of a text file named "sample.txt" and counts the number of words in it. Print the total word count.

d. Create a new text file named "copy.txt." Read the contents of "sample.txt" and write the same contents to "copy.txt." Verify the success of the operation by printing a message.

Question 8:
Dictionaries

a. Create a dictionary representing a person with keys such as name, age, and city. Print the person's information.
b. Create a dictionary with keys representing fruit names and values representing their prices. Print the dictionary and then add a new fruit.
c. Create a dictionary with at least three key-value pairs. Use the keys(), values(), and items() methods to print the keys, values, and key-value pairs.
d. Remove one key-value pair from the dictionary created in the question above using the pop() method. Print the updated dictionary.
e. Given a dictionary, check if a specific key exists. Print a message based on the result.
f. Create a nested dictionary representing information about a person, with keys for "name," "address," and "contact." Print the nested dictionary.

Question 9:
Input Validation

a. Prompt the user to enter a number. Ensure the input is a valid integer and print the square of that number.
b. Ask the user to input their age. If the age is less than 18, print a message saying they are a minor; otherwise, print a message saying they are an adult.
c. Prompt the user to enter two numbers as strings. Convert the strings to integers, add them, and print the result.

Question 10:
Working with Sets

a. Create two sets of your choice. Perform a union and an intersection operation on the sets, and print the results.
b. Create two sets, set1 and set2, each containing at least five elements. Print the union, intersection, and difference of these sets.
c. Create an empty set called my_set. Add three elements to the set and then remove one element. Print the final set.
d. Create two sets and find their symmetric difference. Print the result.
e. Create two sets and check if they are disjoint (have no elements in common). Print the result.

Question 11:
Nested Lists

a. Create a nested list representing a 2x3 matrix. Print the element in the second row and third column.
b. Create two matrices as nested lists. Perform matrix addition (element-wise) and print the result.

c. Create a matrix as a nested list. Write a program to transpose the matrix (swap rows with columns) and print the transposed matrix.

Question 12:
List Comprehension

a. Create a list of the squares of even numbers from 1 to 10 using list comprehension.
b. Generate a list of even numbers between 1 and 20 using list comprehension.
c. Given a list of integers, create a new list with only the positive numbers using list comprehension.
d. Given a list of words, create a new list containing the count of vowels in each word using list comprehension.
e. Create a 3x3 matrix using nested list comprehension. Each element should be the product of its row and column index.
f. Given a matrix (2D list), create a flat list containing all the elements using list comprehension.