

## Practical Session 4

Date: 29/07/2025

Solve the following problems using Jupyter Notebook. Please write the following for each of the programming assignments.

1. The problem statement
2. The entire program
3. The sample input
4. The sample output

Please get the program report signed by the instructor.

### Dictionaries, User Input, While Loops

1. Use a dictionary to store information about a person you know. Store their first name, last name, age, and the city in which they live. You should have keys such as **first\_name**, **last\_name**, **age**, and **city**. Print each piece of information stored in your dictionary. Add, modify, and remove some of the keys and print each piece of information again.
2. Make a dictionary containing three major rivers and the country each river runs through. One key-value pair might be **'nile': 'egypt'**.
  - a. Use a loop to print a sentence about each river, such as **The Nile runs through Egypt**.
  - b. Use a loop to print the name of each river included in the dictionary.
  - c. Use a loop to print the name of each country included in the dictionary.
  - d. Use a loop to print the name of each river in alphabetical order.
3. Ask the user for a number, and then report whether the number is a multiple of 10 or not.
4. A movie theater charges different ticket prices depending on a person's age. If a person is under the age of 3, the ticket is free; if they are between 3 and 12, the ticket is \$10; and if they are over age 12, the ticket is \$15. Write a loop in which you ask users their age, and then tell them the cost of their movie ticket.
5. Write different versions of **Q. No. 4** that do each of the following at least once:
  - a. Use a conditional test in the while statement to stop the loop
  - b. Use an active variable to control how long the loop runs.
  - c. Use a break statement to exit the loop when the user enters a **'quit'** value.

6. Make a list called **sandwich\_orders** and fill it with the names of various sandwiches. Then make an empty list called **finished\_sandwiches**. Loop through the list of sandwich orders and print a message for each order, such as **I made your tuna sandwich**. As each sandwich is made, move it to the list of finished sandwiches. After all the sandwiches have been made, print a message listing each sandwich that was made.
7. Using the list **sandwich\_orders** from Q. No. 6, make sure the sandwich 'pastrami' appears in the list at least three times. Add code near the beginning of your program to print a message saying the deli has run out of pastrami, and then use a while loop to remove all occurrences of 'pastrami' from **sandwich\_orders**. Make sure no pastrami sandwiches end up in **finished\_sandwiches**.