

# Object Oriented Programming (OOP)

## Week-8

### Criteria

Online ordering has enabled many restaurants to manage their peak business hours very effectively. Thanks to online ordering, many people manage to prevent the painful experience of wasting time in a long queue.

AZA is one of the biggest restaurant chains in the United Kingdom. They decided to offer their customers a convenient and contactless mobile ordering solution in response to the growing business need and COVID restrictions.

As part of the development team in the SAS software solutions, you are required to design and develop the mobile ordering program.

Before the application release deadline, you are required to submit the following deliverables:

1. Java program for the two tasks as per the description that follows in this document.

#### Program Description:

In this program you must create a signup process for the mobile ordering application for a restaurant. The restaurant has a variety of cuisines to offer their customers.

When the program starts, the user is given the following options:

1. Sign up
2. Quit Application

#### Output:

Please enter 1 for Sign up. Please enter 2 for Quit.
---

Your program should keep running and enable multiple users to sign up until the quit application option is selected.

To start using the mobile app, users should sign up for an account. You are required to create a Java program for the signup process. The users will be asked to enter their full name, contact number, date of birth, password, and password confirmation.

1. The signup process must not be successful until:
  - a. The full name is entered. The length must be greater than four.
  - b. The mobile number has 10 digits starting with 0.
  - c. The Password must initiate with alphabets followed by either one of @, & and ending with numeric.  
(For Example: John@0125 or John&25) .
  - d. The password confirmation matches the initial entered password.
  - e. The DOB is in the format DD/MM/YYYY or MM/DD/YYYY.
  - f. The user is at least 21 years old. The age should be calculated based on the year entered in the DOB (Only consider year).

**Hint : Use the String Method**

2. If any of the above-mentioned conditions is not fulfilled; the sign-up process should fail, and a descriptive message should be displayed for the user explaining what has gone wrong and providing hints on the correct expected input. The program should keep asking the user to re-enter his details as long as one or more of the input fields are not correctly entered. If all fields are entered successfully, the program should stop asking the user to re-enter his details and display a message that the signup process has been completed successfully.

3. If any field is entered incorrectly, some examples of sample outputs are given below.

Output 1:

You have entered the Date of Birth in invalid format.  
Please start again.

Output 2:

Your passwords do not match.  
Please start again.

Output 3: If User enters 2

Thank you for using the Application.

Output 4: If User enters greater than 2 and less than 1

Invalid Input

4. If all the above-mentioned conditions are successful, the user data is saved in appropriate data structure (Hint: Arrays can be used) to enable data checks during the login process in future builds.

Output 1:

Please enter your full name: Sam Jahn

Please enter your mobile number (username): 0445544455

Please enter your password: John@21

Please confirm your password: John@21

Please enter your Date of Birth #DD/MM/YYYY (No space):  
21/01/1984

You have successfully signed up.

Please enter 1 for Sign up.

Please enter 2 for Quit.