

C-TAG

CODING AND TRAINING ACADEMY BY GANDHIS

Marks: 30

Problem Statement: Banking System Using Classes and Objects in C++

Objective: Create a Banking System in C++ Programming that uses classes and objects. The program should allow users to register, log in, and perform basic banking operations like viewing account details, withdrawing, and depositing money.

Requirements:

1. Class BankAccount:

- **The class should contain the following private members:**
 - **username (string), email (string), phone (string), age (int), balance (double), password (string), accountType (string).**
- **The class should have the following public member functions:**
 - **setDetails(): Takes all user details as input and sets the account details.**
 - **viewAccountDetails(): Displays the account details such as username, email, phone number, account type, and current balance.**
 - **withdraw(): Allows the user to withdraw money from their account. The function should check whether the user has sufficient balance after deducting the minimum balance based on their age.**
 - **deposit(): Allows the user to deposit money into their account, ensuring the deposit amount matches the number of notes entered.**
 - **authenticate(): Compares the username and password for authentication.**

2. Class BankSystem:

- **This class manages user registration and login and provides a menu-driven interface for banking operations.**
- **The first screen should show a welcome message and ask the user to register by entering a username, email, phone, age, and password.**
- **The system should check if the username contains spaces. If it does, prompt the user to enter a valid username.**
- **After successful registration, the details (username and password) should be saved to a file login.dat.**

C-TAG

CODING AND TRAINING ACADEMY BY GANDHIS

Marks: 30

- **The user can log in using their registered username and password.**
- **Once logged in, the user should have the following options:**
 - **View account details.**
 - **Withdraw money (with minimum balance requirements based on age).**
 - **Deposit money.**
 - **Exit.**

Functionality Breakdown:

1. Registration:

- **The user provides their username, email, phone, age, and password. Ensure that the username contains no spaces.**
- **Save the user's login credentials to login.dat.**

2. Login:

- **The user logs in with their username and password. The system verifies the credentials by reading from login.dat.**

3. Account Operations:

- **View Account Details:** The user can view their account details including username, email, phone number, account type, and current balance.
- **Withdraw Money:** The system checks if the user has enough balance considering the minimum balance required based on age, and processes the withdrawal.
- **Deposit Money:** The system allows the user to deposit money by specifying the number of notes of denominations 100, 200, 500, and 2000, and verifies if the total amount matches the input

Functionality Breakdown:

1. Registration:

- The user provides their username, email, phone, age, and password. Ensure that the username contains no spaces.
- Save the user's login credentials to login.dat.

2. Login:

C-TAG

CODING AND TRAINING ACADEMY BY GANDHIS

Marks: 30

- The user logs in with their username and password. The system verifies the credentials by reading from login.dat.

3. Account Operations:

- **View Account Details:** The user can view their account details including username, email, phone number, account type, and current balance.
- **Withdraw Money:** The system checks if the user has enough balance considering the minimum balance required based on age, and processes the withdrawal.
- **Deposit Money:** The system allows the user to deposit money by specifying the number of notes of denominations 100, 200, 500, and 2000, and verifies if the total amount matches the input.

Minimum Balance Requirements:

Age Group	Minimum Balance to be Retained
18 - 25	1000
26 - 45	1500
> 46	2000

Example Input/Output:

Welcome Screen:

```
yaml Copy code

Namaste! Welcome to National Bank!
We hope that you are having a great day.

Please Register to continue:
Enter username: admin_here
Enter Email: admin@nbank.com
Enter Phone: 9999999999
Enter Age: 21
Enter password: *****
Re-Enter password: *****
Type of account: Savings
```

C-TAG

CODING AND TRAINING ACADEMY BY GANDHIS

Marks: 30

Withdraw Example:

```
yaml
Your choice: 2

-----Withdraw Amount-----
Enter the amount to be withdrawn: 4000
Enter your password: ****
...Processing
Success! 4000 withdrawn from your account
Remaining amount: 6000
```

Deposit Example:

```
yaml
Your choice: 3

-----Deposit Amount-----
Current Balance: 6000
Enter the amount to be deposited: 4000
Enter the number of notes of the following:
100: 0
200: 0
500: 4
2000: 1

Success! The amount is deposited to your account
Remaining Balance: 10000
```

Expected Deliverables:

- A well-structured C++ program implementing the banking system.
- The login information should be stored in the login.dat file.
- A screenshot of the contents of the login.dat file after registration.