



[**Git Hub ASSIGEMENT**](#)

// Name: BADAL PRASAD

// Course: B.Tech CSE(Full Stack Development)

// Roll Number: 2501351020

// Assignment: 01

```
import java.util.*;  
class Account {  
    private int accountNumber;  
    private String accountHolderName;  
    private double balance;  
    private String email;  
    private String phoneNumber;  
  
    public Account(int accountNumber, String accountHolderName, double balance, String  
email, String phoneNumber){  
        this.accountNumber = accountNumber;  
        this.accountHolderName = accountHolderName;  
        this.balance = balance;  
        this.email = email;  
        this.phoneNumber = phoneNumber;  
    };  
  
    //Deposit money  
    public void deposit(double amount){  
        if(amount > 0){  
            balance += amount;  
            System.out.println("Balance after deposit: " + balance);  
        }  
        else{  
            System.out.println("Amount should be positive");  
        }  
    };  
  
    //Withdraw money  
    public void withdraw(double amount){
```

```

        if(amount > 0 ){
            if(amount < balance){
                balance -= amount;
                System.out.println("Balance after deposit: "+ balance);
            }
            else{
                System.out.println("Insufficient balance");
            }
        }
        else{
            System.out.println("Amount should be positive");
        }
    }

//Display detail
public void displayAccountDetails(){
    System.out.println("Account number: "+accountNumber);
    System.out.println("Account holder Name: "+accountHolderName);
    System.out.println("Balance: "+balance);
    System.out.println("Email: "+email);
    System.out.println("Phone Number: "+ phoneNumber);
}

//Update details
public void updateContactDetails(String newEmail, String newPhoneNumber){
    this.email = newEmail;
    this.phoneNumber = newPhoneNumber;
    System.out.println("Details updated successfully!!!");
}

//Get account no
public int getAccountNumber() {
    return accountNumber;
}

}

public class BankingApplication{
    Account[] accounts = new Account[100]; // Array of accounts
    int count = 0;                      // Number of accounts created
    Scanner sc = new Scanner(System.in);

    // Create a new account
    public void createAccount() {

```

```
System.out.print("Enter account holder name: ");
String name = sc.nextLine();

System.out.print("Enter initial deposit amount: ");
double amount = sc.nextDouble();
sc.nextLine(); // clear input buffer

System.out.print("Enter email: ");
String email = sc.nextLine();

System.out.print("Enter phone number: ");
String phone = sc.nextLine();

int accNum = 1000 + count + 1; // simple auto account number

Account newAcc = new Account(accNum, name, amount, email, phone);
accounts[count] = newAcc;
count++;

System.out.println("Account created successfully with Account Number: " + accNum);
}

// Deposit money
public void performDeposit() {
    System.out.print("Enter account number: ");
    int accNum = sc.nextInt();
    sc.nextLine();

    Account acc = findAccount(accNum);
    if (acc == null) {
        System.out.println("Account not found!");
        return;
    }

    System.out.print("Enter amount to deposit: ");
    double amount = sc.nextDouble();
    sc.nextLine();

    acc.deposit(amount);
}

// Withdraw money
public void performWithdrawal() {
    System.out.print("Enter account number: ");
```

```
int accNum = sc.nextInt();
sc.nextLine();

Account acc = findAccount(accNum);
if (acc == null) {
    System.out.println("Account not found!");
    return;
}

System.out.print("Enter amount to withdraw: ");
double amount = sc.nextDouble();
sc.nextLine();

acc.withdraw(amount);
}

// Show account details
public void showAccountDetails() {
    System.out.print("Enter account number: ");
    int accNum = sc.nextInt();
    sc.nextLine();

    Account acc = findAccount(accNum);
    if (acc == null) {
        System.out.println("Account not found!");
    } else {
        acc.displayAccountDetails();
    }
}

// Update contact details
public void updateContact() {
    System.out.print("Enter account number: ");
    int accNum = sc.nextInt();
    sc.nextLine();

    Account acc = findAccount(accNum);
    if (acc == null) {
        System.out.println("Account not found!");
        return;
    }

    System.out.print("Enter new email: ");
    String email = sc.nextLine();
```

```

System.out.print("Enter new phone number: ");
String phone = sc.nextLine();

acc.updateContactDetails(email, phone);
}

// Find account by account number
private Account findAccount(int accNum) {
    for (int i = 0; i < count; i++) {
        if (accounts[i].getAccountNumber() == accNum) {
            return accounts[i];
        }
    }
    return null;
}

// Main Menu
public void mainMenu() {
    while (true) {
        System.out.println("\nWelcome to the Banking Application!");
        System.out.println("1. Create a new account");
        System.out.println("2. Deposit money");
        System.out.println("3. Withdraw money");
        System.out.println("4. View account details");
        System.out.println("5. Update contact details");
        System.out.println("6. Exit");

        System.out.print("Enter your choice: ");
        int choice = sc.nextInt();
        sc.nextLine();
        switch (choice) {
            case 1: createAccount(); break;
            case 2: performDeposit(); break;
            case 3: performWithdrawal(); break;
            case 4: showAccountDetails(); break;
            case 5: updateContact(); break;
            case 6:
                System.out.println("Thank you for using the Banking Application!");
                return;
            default:
                System.out.println("Invalid choice!");
        }
    }
}

```

```
}

public static void main(String[] args) {
    BankingApplication ba = new BankingApplication();
    ba.mainMenu();
}
}
```