

## CODSOFT-TASK3(ATM INTERFACE)

**Ashmitha P** <ashmithaparama@gmail.com>
To: Ashmitha P

Ash		

import java.util.Scanner; public class ATM {
// The user's bank account
public BankAccount bankAccount;
// Constructor
public ATM(BankAccount bankAccount) {     this.bankAccount = bankAccount;
}
// Withdraws cash from the ATM
public void withdraw(){
public volu withdraw()
System.out.println("Enter the amount you want to withdraw: ");  Scanner scanner = new Scanner(System.in);
Scanner scanner = new Scanner(System.in);
int amount = scanner.nextInt();
withdraw(amount);
<u>,</u>
}
// Deposits cash into the ATM
nublic void denosit () [
public void deposit() {
System.out.println("Enter the amount you want to deposit: "); Scanner scanner = new Scanner(System.in);
Scanner scanner = new Scanner(System.in);
int amount = scanner.nextInt();
deposit(amount);
}
// Withdraws cash from the account
private void withdraw(int amount) {  if (bankAccount.getBalance() < amount) {
if (bankAccount.getBalance() < amount) {
System.out.println("Insufficient funds");
}
else
bankAccount.withdraw(amount);
System.out.println("Amount Withdraw Sucessfully "+amount);
}
}
// Deposits cash into the account
// Deposits cash into the account private void deposit(int amount) {
bankAccount.deposit(amount);
System.out.println("Amount Deposited Sucessfully "+amount);
}
// Checks the account balance
// Checks the account balance private void checkBalance() {
private void cneckBalance() {
System.out.println("Your account balance is: "+bankAccount.getBalance());
}
public static void main(String[] args){
BankAccount bankAccount = new BankAccount(1000);
ATM atm = new ATM(bankAccount);

ii	int choice;	
	Scanner scanner = new Sca	nner(System.in);
(	do	
1	{	
	System.out.println("ATM Me	nu"):
	System.out.println("1. Witho	raw );
	System.out.println("2. Depo	sit");
	System.out.println("3. Check	. Dalace.").
	System.out.println("4. Exit");	
	System.out.println("Please e	enter your choice: ");
	choice = scanner.nextInt();	
	- Searmer Hextine(),	
	switch (choice) {	
(	case 1:	
ć	atm.withdraw();	
	break;	
(	case 2:	
	atm.deposit();	
	break;	
	case 3:	
	case 3:	
	atm.checkBalance();	
}	break;	
(	case 4:	
5	System.exit(0);	
	la constitución	
	break;	
(	default:	
	Cuetara out printla ("lovalid a	heise").
	System.out.println("Invalid c	noice ,
)	}	
)		
	}while(choice!=4);	
	}while(choice!=4); scanner.close();	
) (		
)		
)		
)	scanner.close(); } }	
)      		
)	scanner.close(); }  public class	
)	scanner.close(); } }	
)	scanner.close(); }  public class	
)	scanner.close(); }  public class	
	scanner.close(); }  public class	
)	scanner.close(); }  public class	// The account balance
)	scanner.close(); }  public class	// The account balance private int balance;
	scanner.close(); }  public class	// The account balance
	scanner.close(); }  public class	// The account balance private int balance;
)	scanner.close(); }  public class	// The account balance private int balance; // Constructor
)	scanner.close(); }  public class	// The account balance private int balance; // Constructor
)	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) (
	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) { this.balance = balance:
	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) (
	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) { this.balance = balance;
	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this balance = balance; }
)	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this.balance = balance; }  // Withdraws cash from the account
)	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this balance = balance; }  // Withdraws cash from the account
	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this.balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {
)	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this.balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {
)	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {
)	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this.balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System.out.println("Insufficient funds");     }
)	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {
)	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this.balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System.out.println("Insufficient funds");     }
	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this.balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System.out.println("Insufficient funds");     }
	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this.balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System.out.println("Insufficient funds");     }      balance == amount;
	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this.balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System.out.println("Insufficient funds");     }  balance == amount:
	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this.balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System.out.println("Insufficient funds");     }      balance == amount;
	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this.balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System.out.println("Insufficient funds");     }      balance == amount; }
)	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this.balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System.out.println("Insufficient funds");     }  balance == amount; }  // Deposits cash into the account
)	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System.out.println("Insufficient funds");     }      balance == amount; }  // Deposits cash into the account
)	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this.balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System.out.println("Insufficient funds");     }  balance -= amount; }  // Deposits cash into the account public void deposit(int amount) {
)	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System.out.println("Insufficient funds");     }      balance -= amount; }  // Deposits cash into the account public void deposit(int amount) {         balance += amount; }
)	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System.out.println("Insufficient funds");     }      balance -= amount; }  // Deposits cash into the account public void deposit(int amount) {     balance += amount; }
)	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this.balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System.out.println("Insufficient funds");     }      balance -= amount; }  // Deposits cash into the account public void deposit(int amount) {     balance += amount;
	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this.balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount, intin("insufficient funds"); }  balance -= amount; }  // Deposits cash into the account public void deposit(int amount) {     balance += amount; }
) lamanananani.	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System.out.println("Insufficient funds");     }      balance -= amount; }  // Deposits cash into the account public void deposit(int amount) {     balance += amount; }
) lannananananananananananananananananana	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System out printin("insufficient funds");     }  balance == amount; }  // Deposits cash into the account public void deposit(int amount) {     balance += amount; }  // Checks the account balance
	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System.out.println("Insufficient funds");     }      balance == amount; }  // Deposits cash into the account public void deposit(int amount) {     balance += amount; }  // Checks the account balance public int getBalance() {
innamental.	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System out printin("insufficient funds");     }  balance == amount; }  // Deposits cash into the account public void deposit(int amount) {     balance += amount; }  // Checks the account balance
) immanumum.	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this.balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System.out.println('Insufficient funds');     }      balance amount; }  // Deposits cash into the account public void deposit(int amount) {     balance += amount; }  // Checks the account balance public int getBalance() {     return balance;
) innumumani.	scanner.close(); }  public class	// The account balance private int balance;  // Constructor public BankAccount(int balance) {     this balance = balance; }  // Withdraws cash from the account public void withdraw(int amount) {     if (balance < amount) {         // System.out.println("Insufficient funds");     }      balance == amount; }  // Deposits cash into the account public void deposit(int amount) {     balance += amount; }  // Checks the account balance public int getBalance() {