pseudoCode

Import Scanner  
Import Date

Start class

Static variables   
 Scanner - input  
 int – Index  
 boolean PassChangeSure = false, lodraw  
 int arrays - PIN, Balance  
 boolean array -  Overdraft  
 String array – Name  
   
  
 Start main method  
 Variables – Date d  
 Print Date and welcome message  
 Call Login() method  
 End main method

Start login() method  
 Variables  
 String PINStr  
 int PIN

flag=false

Prompt user and get a pin from the them as a string (PINStr)

            Pass the PIN string to the ValidatePIN() method  
 The validated PIN is returned as int  
 loop to search for Pin

for(int i = 0;i<pin.length;i++)  
 Store the  position of the user in the Array

Use boolean flag to check if true or false

if(flag == true){  
 Welcome the user by their name  
 Call  Display() method}  
End Login() Method

Start displayMenu()  
 Variable String – ChoiceStr int - Choice   
 Show all options for the user (1-5)   
 Prompt the user for ChoiceStr (option 1-5)  
 Call the Validatechoice() Method and pass ChoiceStr to it  
 The choice int is returned.  
 Use 5 “If” Statements to choose an option calling the correct methods in each.  if(choice==1){

userStatement;

}

End DisplayMenu() Method

Start userStatement() method  
 Print Pin, Name, Balance  
 Check if overdraft facility is available and print if it is or isnt  
 Call displayMenu  
end userStatement method

Start lodge() method  
 Declare variables - String lodgestr \_ int lodgeAmount  
 Set the static boolean lodraw to true. Used in validation  
 Prompt the user for the amount to lodge. Store in lodgestr string.  
 Send lodgestr to the validatePosiDigits() method. lodgeAmount is returned.  
 Put the users balance = to balance + lodgeAmount  
 Print the users new balance  
 Call displayMenu method  
 End lodge() method

Start withdraw() method  
 Declare variables - String withdrawstr \_ int withdrawAmount  
 Set the static boolean lodraw to false. Used in validation.  
 Prompt the user for the amount to withdraw. Store as the withdrawstr String.  
 Send withdrawstr to validatePosiDigits(). withdrawAmount is returned  
 If the Amount is greater then the balance and the user cannot overdraft print  
                            an error to the user and recall the withdraw() method  
 Put the users balance = to balance + withdrawAmount  
 Print the users new balance.  
 Call the displayMenu() method.  
 End withdraw() method

Start changePassword() method  
 Declare variables  String userSure = “”,newPinstr;int newPin;  
 set PassChangeSure to false.  
 If PassChangeSure == false  
 Prompt the user to see if they are sure to change password   
 input = userSure  
 End if  
 else if PassChangeSure == true  
 set userSure = “y”  
 end else if  
 Reset ChangePassSure for confirmation in future PINs  
 if userSure == y || Y  
 Prompt the user for the new PIN. Store as String.  
 Send the PIN String to be validated in the validatePin method  
 The PIN int is returned from the method.  
 Use a for loop to check that the PIN is not in use. (i=0; i<pinlength; i++)  
 if PIN(0) == newPin  
 Print that the PIN is in use  
 Set PassChangeSure to true  
 Recall the method changePassword()  
 End if  
 End for loop  
 Pin(index) = newPin;  
 Call displayMenu() method  
 End if  
 Else if userSure == n || N  
 Print “Returning to menu”  
 Call displayMenu() method  
 End else if  
 else  
 Print error message to user “Please input N for No or Y for Yes”  
 recall changePassword() method  
 end else

End changePassword() method

Start pinValidation() Method  
 while loop  
 validate, check for non-digits and that its three digits long  
 Prompt user to re-enter digits and check again  
 take in the input  
 end loop  
 return pin   
 End PINValidation() Method

Start validateMenu() method  
 while loop  
 validate, check for non-digits, if its between1-6  
 Prompt user re-enter digits and check again  
 take in the input  
 end loop  
 return choice  
 end choice method

Start validatePosDigits() method  
 while loop  
 validate, check for non-digits  
 Prompt user re-enter lodgement if boolean lodraw is positive  
 If lodraw is false then prompt user for withdrawal amount  
 Take in the input  
 end loop  
 return choice  
 End validatePosDigits method

End Class