**SDL ASSIGNMENT 1**

NAME : Bhavik Ransubhe

CLASS: TE (B) COMP

ROLL NO:39055

**1.Title:**

Java system with advance data structures system using collections and generics.

**2. Objective:**

To design a system with the help of advance data structures in Java .

To enhance the system using collections and generics.

**3.PROBLEM STATEMENT :**

Design a system with the help of advance data structures in Java and enhance the system using collections and generics.

**4. Outcomes:**

We will be able to use advanced data structures in Java.

We will also be able to apply collections and generics as per our applications.

**5. Software & Hardware Requirements:**

Programming tools recommended**:** Eclipse JAVA / Intellij IDEA .

PC/LAPTOP with windows 10 ,min 4GB RAM , 500GB HDD , Intel Core i5 Processor.

**6. Theory Concepts in brief:**

The ArrayList class extends AbstractList and implements the List interface. ArrayList supports dynamic arrays that can grow as needed.

Standard Java arrays are of a fixed length. After arrays are created, they cannot grow or shrink, which means that you must know in advance how many elements an array will hold.

Array lists are created with an initial size. When this size is exceeded, the collection is automatically enlarged. When objects are removed, the array may be shrunk.

**Declaration:**

* List< Type > object\_name = new LinkedList< Type >()

**Methods:**

* **add(element) / add (index, element):**

Inserts the specified element at the specified position index in this list. Throws IndexOutOfBoundsException if the specified index is out of range(index < 0 || index > size()).

* **addAll(Collection):**

Inserts all of the elements in the specified collection into this list, starting at the specified position. Throws NullPointerException if the specified collection is null.

* **get(index):**

Returns the element at the specified position in this list. Throws IndexOutOfBoundsException if the specified index is out of range (index < 0 || index >=size ()).

* **set(index,value):**

Replaces the element at the specified position in this list with the specified element. Throws IndexOutOfBoundsException if the specified index is out of range (index < 0 || index >= size ()).

* **remove(index):**

Removes the element at the specified position in this list. Throws IndexOutOfBoundsException if the index out is of range (index < 0 || index >= size ()).

* **removeAll ():**

Removes from this list all of its elements that are contained in the specified collection.

* **Clear ():**

Removes all of the elements from this list.

* **Size ():**

Returns the number of elements in this list.

* **Contains ():**

Returns true if this list contains the specified element. More formally, returns true if and only if this list contains at least one element e such that (o==null ? e==null : o.equals(e)).

* **isEmpty ():**

Returns true if this list contains no elements.

* **toArray ():**

Returns an array containing all of the elements in this list in the correct order. Throws NullPointerException if the specified array is null.

**7. Features of tool/framework/language used:**

The prime reason behind creation of Java was to bring portability and security features into a computer language. Beside these two major features, there were many other features that played an important role in moulding out the final form of this outstanding language

In Java, everything is an Object. Java can be easily extended since it is based on the Object model.

#### **Platform Independent**

Unlike many other programming languages including C and C++, when Java is compiled, it is not compiled into a platform specific machine, rather into platform-independent bytecode. This byte code is distributed over the web and interpreted by the Virtual Machine (JVM) on whichever platform it is being run on.

#### **Simple**

Java is designed to be easy to learn. If you understand the basic concept of OOP Java, it would be easy to master.

#### **Secure**

With Java's secure features it enables to develop virus-free, tamper-free systems. Authentication techniques are based on public-key encryption.

#### **Architecture-neutral**

Java compiler generates an architecture-neutral object file format, which makes the compiled code executable on many processors, with the presence of a Java runtime system.

#### **Portable**

Being architecture-neutral and having no implementation dependent aspects of the specification makes Java portable. The compiler in Java is written in ANSI C with a clean portability boundary, which is a POSIX subset.

#### **Robust**

Java makes an effort to eliminate error-prone situations by emphasizing mainly on compile time error checking and runtime checking.

#### **Multithreaded**

With Java's multithreaded feature it is possible to write programs that can perform many tasks simultaneously. This design feature allows the developers to construct interactive applications that can run smoothly.

#### **Interpreted**

Java byte code is translated on the fly to native machine instructions and is not stored anywhere. The development process is more rapid and analytical since the linking is an incremental and light-weight process.

#### **High Performance**

With the use of Just-In-Time compilers, Java enables high performance.

#### **Distributed**

Java is designed for the distributed environment of the internet.

#### 

#### **Dynamic**

Java is considered to be more dynamic than C or C++ since it is designed to adapt to an evolving environment. Java programs can carry an extensive amount of run-time information that can be used to verify and resolve accesses to objects at run-time.

**8. Conclusion:** We have successfully implemented advanced data structures in Java and also were able to apply collections and generics as per our applications.

* **PROGRAM CODE (Bank System) :-**

**FILE 1: Main.java:**

package sdl\_Assign;

import java.text.SimpleDateFormat;

import java.util.ArrayList;

import java.util.Scanner;

public class Main {

public static ArrayList<accountData>accountDataArrayList;

public static Scanner scanner;

public static void main(String[] args) {

accountDataArrayList = new ArrayList<>();

scanner = new Scanner(System.in);

int ch;

do {

System.out.print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\nBR14x Bank System\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n1-Login User\n2-Create an Account\n3-Deactivate Account\n4-Show Bank Accounts\n5-Exit \n : ");

ch = scanner.nextInt();

switch (ch) {

case 1: **//IF ACCOUNT ALREADY EXITS**

System.out.print("Enter Account Number:");

int accNo = scanner.nextInt();

scanner.nextLine();

System.out.print("Enter Password :");

String pass = scanner.nextLine();

boolean stat = false;

int index = 0;

for (int i = 0; i < accountDataArrayList.size(); i++) {

if (accountDataArrayList.get(i).getAccountNumber() ==

accNo) {

stat = true;

index = i;

break;

}

}

if (stat) {

String pass1 =

accountDataArrayList.get(index).getAccountPassword();

if (pass1.equals(pass)) {

System.out.println("WELCOME TO BR14x SALISBURY BRANCH BANK");

showInformation(2,index);

loggedInUser(index);

} else {

System.out.println("Wrong Password");

break;

}

} else {

System.out.println("Account Not Found");

}

System.out.println("Pass :" + pass);

break;

case 2: **//CREATING NEW ACCOUNT**

accountData account = new accountData();

account.getData();

System.out.println("Account Created");

accountDataArrayList.add(account);

break;

case 3:

System.out.print("Enter Account Number :");

accNo = scanner.nextInt();

scanner.nextLine();

System.out.print("Enter Password :");

pass = scanner.nextLine();

stat = false;

index = 0;

for (int i = 0; i < accountDataArrayList.size(); i++) {

if (accountDataArrayList.get(i).getAccountNumber() ==

accNo) {

stat = true;

index = i;

break;

}

}

if (stat) {

String pass1 =

accountDataArrayList.get(index).getAccountPassword();

if (pass1.equals(pass)) {

System.out.print("Do you really want to delete your account ?[Y/N]");

String ch1 = scanner.next();

if (ch1.equals("Y")) {

accountDataArrayList.remove(index);

System.out.println("Account Deleted\nThanks for banking with us!");

} else {

System.out.println("Account Not Deleted");

}

} else {

System.out.println("Wrong Password");

break;

}

} else {

System.out.println("Account Not Found");

}

break;

case 4:

System.out.print("Enter Master Password :");

int passM = scanner.nextInt();

if (passM == 9669) {

System.out.println("Loading Data");

} else {

System.out.println("Wrong Password");

}

showInformation(1,0);

break;

}

} while (ch < 5);

}

public static void loggedInUser(int index) {

int ch;

do { **//MENU ONCE CUSTOMER IS LOGGED IN**

System.out.print("\n| CUSTOMER PORTAL | \n1-Show Account Details \n2-Withdraw Amount\n3-Deposit Amount\n4-Transfer Amount\n5-Update Account Details\n6-Show Statement\n7-LogOut : ");

ch = scanner.nextInt();

switch (ch) {

case 1:

showInformation(2,index);

break;

case 2:

withdraw(index);

break;

case 3:

deposit(index);

break;

case 4:

transferAmount(index);

break;

case 5:

updateData(index);

break;

case 6:

showStatement(index);

break;

}

} while (ch < 7);

}

public static void transferAmount(int indexSender) {

int accNo;

boolean stat = false;

int indexReceiver = 0;

double amount;

System.out.println("TRANSFER MONEY PORTAL");

System.out.print("Enter The Name of Account Holder :");

scanner.nextLine();

String Name = scanner.nextLine();

System.out.print("Enter The Account Number Of Holder To Transfer Amount:");

accNo = scanner.nextInt();

for (int i = 0; i < accountDataArrayList.size(); i++) {

if (accountDataArrayList.get(i).getAccountNumber() == accNo &&

Name.equals(accountDataArrayList.get(i).getUserName())) {

stat = true;

indexReceiver = i;

break;

}

}

if (stat) {

scanner.nextLine();

System.out.print("Enter Amount to Transfer :");

amount = scanner.nextDouble();

double Amt1 =

accountDataArrayList.get(indexSender).getAccountBalance();

double Amt2 =

accountDataArrayList.get(indexReceiver).getAccountBalance();

if (Amt1 > amount + 3000.0) {

Amt1 -= amount;

Amt2 += amount;

accountDataArrayList.get(indexSender).setAccountBalance(Amt1);

accountDataArrayList.get(indexReceiver).setAccountBalance(Amt2);

SimpleDateFormat formattedDate = new SimpleDateFormat("dd:MM:yyyy hh:mm a");

String date = formattedDate.format(System.currentTimeMillis());

transaction Trans = new transaction(date, amount, "Withdraw",amount, "Transferred To " +

accountDataArrayList.get(indexReceiver).getUserName());

accountDataArrayList.get(indexSender).transactionStatement.add(Trans);

transaction Trans1 = new transaction(date, amount, "Deposit",

amount, "Transferred From " +

accountDataArrayList.get(indexSender).getUserName());

accountDataArrayList.get(indexReceiver).transactionStatement.add(Trans1);

} else {

System.out.print("You Dont Have Enough Balance To Transfer The Money");

}

} else {

System.out.println("Wrong Account Details");

}

}

public static void withdraw(int index) {

double amount = accountDataArrayList.get(index).getAccountBalance();

double withdrawAmount;

System.out.print("Enter The Amount You Withdraw : ");

withdrawAmount = scanner.nextDouble();

if (withdrawAmount > amount) {

System.out.println("Account Balance is low to withdraw amount " +

withdrawAmount);

} else {

amount -= withdrawAmount;

System.out.println("Amount Withdrawn \nAccount Balance : " + amount);

accountDataArrayList.get(index).setAccountBalance(amount);

SimpleDateFormat formattedDate = new SimpleDateFormat("dd:MM:yyyy hh:mm a");

String date = formattedDate.format(System.currentTimeMillis());

transaction Trans = new transaction(date, withdrawAmount, "Withdraw",

amount, "Withdrawn Self");

accountDataArrayList.get(index).transactionStatement.add(Trans);

}

}

public static void deposit(int index) {

double amount = accountDataArrayList.get(index).getAccountBalance();

double depositAmount;

System.out.print("Enter The Amount You Deposit : ");

depositAmount = scanner.nextDouble();

amount += depositAmount;

System.out.println("Amount Deposited \nAccount Balance :" + amount);

accountDataArrayList.get(index).setAccountBalance(amount);

SimpleDateFormat formattedDate = new SimpleDateFormat("dd:MM:yyyy hh:mm a");

String date = formattedDate.format(System.currentTimeMillis());

transaction Trans = new transaction(date, depositAmount, "Deposit",

amount, "Deposited Self");

accountDataArrayList.get(index).transactionStatement.add(Trans);

}

public static void showStatement(int index) {

for (int i = 0; i < 110; i++)

System.out.print('\_');

System.out.println();

System.out.format("%1$-21s%2$-10s%3$-12s%4$-10s%5$-21s",

"Date", "Amount", "Type", "Balance", "Description");

System.out.println();

for (int i = 0; i < 110; i++)

System.out.print('\_');

System.out.println();

for (int i = 0; i <

accountDataArrayList.get(index).transactionStatement.size(); i++) {

if(i!=0) {

System.out.println();

}

transaction trans =

accountDataArrayList.get(index).transactionStatement.get(i);

System.out.format("%1$-21s%2$-10s%3$-12s%4$-10s%5$-21s",

trans.getDate(),

trans.getAmount(),

trans.getType(),

trans.getTotalBalance(),

trans.getDescription());

}

System.out.println();

for (int i = 0; i < 110; i++)

System.out.print('\_');

}

public static void updateData(int index) { **//FUNCTION TO UPDATE/CHANGE DETAILS**

int ch;

String field2;

int ch2;

System.out.print("\nEnter Field To Change\n1.Name\n2.DOB\n3.PAN\n4.Address\n5.Nominee\n6.PhoneNumber\n7.Aadhar No");

ch = scanner.nextInt();

switch (ch) {

case 1:

scanner.nextLine();

System.out.println("Current Name :"

+accountDataArrayList.get(index).getUserName());

System.out.print("Enter Change Name To :");

field2 = scanner.nextLine();

System.out.print("Change Name [1/0]?");

ch2 = scanner.nextInt();

if (ch2 == 1) {

accountDataArrayList.get(index).setUserName(field2);

} else {

System.out.print("Name Unchanged");

}

break;

case 2:

scanner.nextLine();

System.out.println("Current DOB :"

+accountDataArrayList.get(index).getUserDOB());

System.out.print("Enter Change DOB To :");

field2 = scanner.nextLine();

System.out.print("Change DOB [1/0]?");

ch2 = scanner.nextInt();

if (ch2 == 1) {

accountDataArrayList.get(index).setUserDOB(field2);

} else {

System.out.print("DOB Unchanged");

}

break;

case 3:

scanner.nextLine();

System.out.println("Current PAN :"

+accountDataArrayList.get(index).getUserPAN());

System.out.print("Enter Change PAN To :");

field2 = scanner.nextLine();

System.out.print("Change PAN [1/0]?");

ch2 = scanner.nextInt();

if (ch2 == 1) {

accountDataArrayList.get(index).setUserPAN(field2);

} else {

System.out.print("PAN Unchanged");

}

break;

case 4:

scanner.nextLine();

System.out.println("Current Address :"

+accountDataArrayList.get(index).getUserAddress());

System.out.print("Enter Change Address :");

field2 = scanner.nextLine();

System.out.print("Change Address [1/0]?");

ch2 = scanner.nextInt();

if (ch2 == 1) {

accountDataArrayList.get(index).setUserAddress(field2);

} else {

System.out.print("Address Unchanged");

}

break;

case 5:

scanner.nextLine();

System.out.println("Current Nominee :"

+accountDataArrayList.get(index).getNominee());

System.out.print("Enter Change Nominee :");

field2 = scanner.nextLine();

System.out.print("Change Nominee [1/0]?");

ch2 = scanner.nextInt();

if (ch2 == 1) {

accountDataArrayList.get(index).setNominee(field2);

} else {

System.out.print("Nominee Unchanged");

}

break;

case 6:

scanner.nextLine();

System.out.println("Current Phone Number :"

+accountDataArrayList.get(index).getPhoneNumber());

boolean stat=false;

do {

System.out.print("Enter Change Phone Number To : ");

String phoneNum = scanner.nextLine();

if(phoneNum.length()!=10){

System.out.print("Phone Number Is Not Valid");

System.out.println();

stat=true;

phoneNum=null;

}

else{

for(int i=0;i<10;i++){

assert phoneNum != null;

if(phoneNum.charAt(i) >='0'&&

phoneNum.charAt(i)<='9'){

System.out.print("Change Phone [1/0]?");

ch2 = scanner.nextInt();

if (ch2 == 1) {

accountDataArrayList.get(index).setPhoneNumber(phoneNum);

} else {

System.out.print("Phone Number Unchanged");

}

stat=false;

}

else{

System.out.print("Phone Number Should Not Contain Any Letters");

System.out.println();

phoneNum=null;

stat=true;

}

}

}

}while(stat);

break;

case 7:

scanner.nextLine();

System.out.println("Current Aadhar :"

+accountDataArrayList.get(index).getUserAadhar());

stat=false;

do {

System.out.print("Enter Aadhar : ");

String userAadhar=scanner.nextLine();

if(userAadhar.length()!=12){

System.out.print("Aadhar Number Is Not Valid.");

System.out.println();

stat=true;

userAadhar=null;

}

else{

for(int i=0;i<12;i++){

if(userAadhar.charAt(i) >='0' &&

userAadhar.charAt(i)<='9'){

System.out.print("Change Aadhar [1/0]?");

ch2 = scanner.nextInt();

if (ch2 == 1) {

accountDataArrayList.get(index).setUserAadhar(userAadhar);

} else {

System.out.print("Aadhar Number Unchanged");

}

stat=false;

}

else{

System.out.print("Aadhar Should Not Contain Any Letters");

System.out.println();

userAadhar=null;

stat=true;

}

}

}

}while(stat);

break;

}

}

public static void showInformation(int type, int index) { //1=whole data //2=single

for (int i = 0; i < 155; i++)

System.out.print('\_');

System.out.println();

System.out.format("%1$-13s%2$-22s%3$-15s%4$-15s%5$-12s%6$-12s%7$-30s%8$-20s%9$-17s",

"Account No", "Name", "Phone Number", "Aadhar No", "PAN No",

"DOB", "Address", "Nominee","Account Balance");

System.out.println();

for (int i = 0; i < 155; i++)

System.out.print('\_');

System.out.println();

if(type==1){

for (int i = 0; i < accountDataArrayList.size(); i++) {

if(i!=0) {

System.out.println();

}

show(i);

if(i != accountDataArrayList.size()-1){

for ( int j = 0; j<155; j++)

System.out.print('\_');

}

}

}else if(type==2 ) {

show(index);

}

for (int i = 0; i < 155; i++)

System.out.print('\_');

}

private static void show(int i) { **//function for showing data**

System.out.format("%1$-13s%2$-22s%3$-15s%4$-15s%5$-12s%6$-12s%7$-30s%8$-20s%9$-17s",

accountDataArrayList.get(i).getAccountNumber(),

accountDataArrayList.get(i).getUserName(),

accountDataArrayList.get(i).getPhoneNumber(),

accountDataArrayList.get(i).getUserAadhar(),

accountDataArrayList.get(i).getUserPAN(),

accountDataArrayList.get(i).getUserDOB(),

accountDataArrayList.get(i).getUserAddress(),

accountDataArrayList.get(i).getNominee(),

accountDataArrayList.get(i).getAccountBalance());

System.out.println();

}

}

**FILE 2: transaction.java:**

package sdl\_Assign;

public class transaction {

private String date;

private double amount;

private String type;

private double totalBalance;

private String description;

public double getAmount() {

return amount;

}

public String getDate() {

return date;

}

public transaction(String date, double amount, String type, double totalBalance,String description) {

this.date = date;

this.amount = amount;

this.type = type;

this.totalBalance = totalBalance;

this.description =description;

}

public String getType() {

return type;

}

public double getTotalBalance() {

return totalBalance;

}

public String getDescription() {

return description;

}

}

**FILE 3: accountData.java:**

package sdl\_Assign;

import java.util.ArrayList;

import java.util.Random;

import java.util.Scanner;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

public class accountData {

private String userName,userDOB,userPAN,userAddress,nominee,accountPassword;

private String phoneNumber,userAadhar;

private int accountNumber;

private double accountBalance;

Scanner scanner=new Scanner(System.in);

public ArrayList<transaction> transactionStatement;

private static int latestAccNo = 369800;

public static int generateAccNo() {

return ++latestAccNo;

}

public void getData(){ **//FUNCTION FOR GETTING CUSTOMER DATA**

boolean stat = false;

System.out.println("Welcome , Please fill the details to open your account\n");

Random rand = new Random();

System.out.print("Enter Full Name : ");

userName=scanner.nextLine();

System.out.print("Enter Date Of Birth : ");

userDOB=scanner.nextLine();

System.out.print("Enter Address : ");

userAddress=scanner.nextLine();

System.out.print("Enter Nominee : ");

nominee = scanner.nextLine();

System.out.print("Enter PAN : ");

userPAN = scanner.nextLine();

do {

System.out.print("Enter Phone Number : ");

phoneNumber = scanner.nextLine();

if(phoneNumber.length()!=10){ **//GIVING CONDITIONS FOR VAILD PHONE NUMBER**

System.out.print("Phone Number Is Not Valid");

System.out.println();

stat=true;

phoneNumber=null;

}

else{

for(int i=0;i<10;i++){

assert phoneNumber != null;

if(phoneNumber.charAt(i) >='0' && phoneNumber.charAt(i)<='9'){

stat=false;

}

else{

System.out.print("Phone Number Should Not Contain Any Letters ");

System.out.println();

phoneNumber=null;

stat=true;

}

}

}

}while(stat);

do {

System.out.print("Enter Aadhar : ");

userAadhar=scanner.nextLine();

if(userAadhar.length()!=12){ **//GIVING CONDITIONS FOR VAILD AADHAR** **NUMBER**

System.out.print(" Aadhar Number Is Not Valid.");

System.out.println();

stat=true;

userAadhar=null;

}

else{

for(int i=0;i<12;i++){

if(userAadhar.charAt(i) >='0' && userAadhar.charAt(i)<='9'){

stat=false;

}

else{

System.out.print("Aadhar Should Not Contain Any Letters");

System.out.println();

userAadhar=null;

stat=true;

}

}

}

}while(stat);

do {

System.out.print("Enter Initial Balance (Must Be Above Rs.5000) : ");

accountBalance = scanner.nextDouble();

}while(accountBalance<5000);

System.out.print("Creating An Account.....");

accountNumber = generateAccNo();

scanner.nextLine();

boolean stat1;

String pass1;

**//GIVING PASSWORD CONDIITIONS FOR UNIQUENESS**

do {

System.out.print("\nPlease Enter New Password \nShould be of minimum size 8 with Letter, Digit And Special Characters\n(Remember For Next Login) : ");

pass1 = scanner.nextLine();

if (pass1.length() >= 8) {

Pattern letter = Pattern.compile("[a-zA-z]");

Pattern digit = Pattern.compile("[0-9]");

Pattern special = Pattern.compile("[!@#$%&\*()\_+=|<>?{}\\[\\]~-]");

//Pattern eight = Pattern.compile (&quot;.{8}&quot;);

Matcher hasLetter = letter.matcher(pass1);

Matcher hasDigit = digit.matcher(pass1);

Matcher hasSpecial = special.matcher(pass1);

if (hasDigit.find() && hasLetter.find() && hasSpecial.find()) {

stat1=false;

} else {

System.out.println("Should be of minimum size 8 with LetterDigit And Special Characters");

pass1 = null;

stat1=true;

}

} else {

System.out.println(" Invalid Pass. Should be of minimum size 8 ");

stat1=true;

}

}while (stat1);

System.out.print("Please Enter Password Again To Confirm :");

String pass2 = scanner.nextLine();

if (pass1.isEmpty()) {

System.out.println("Password cant be empty");

} else {

if (pass2.isEmpty()) {

System.out.println("Password cant be empty");

} else {

if (pass1.equals(pass2)) {

accountPassword = pass1;

System.out.println("Account Created With Account Number :"+ accountNumber +"\nWith Password :" + accountPassword);

} else {

System.out.println("Password Do Not Match");

}

}

}

transactionStatement=new ArrayList<>();

}

public String getUserName() {

return userName;

}

public String getUserDOB() {

return userDOB;

}

public String getUserPAN() {

return userPAN;

}

public String getUserAddress() {

return userAddress;

}

public String getNominee() {

return nominee;

}

public String getAccountPassword() {

return accountPassword;

}

public String getPhoneNumber() {

return phoneNumber;

}

public String getUserAadhar() {

return userAadhar;

}

public long getAccountNumber() {

return accountNumber;

}

public double getAccountBalance() {

return accountBalance;

}

public void setAccountBalance(double accountBalance) {

this.accountBalance = accountBalance;

}

public void setUserName(String userName) {

this.userName = userName;

}

public void setUserDOB(String userDOB) {

this.userDOB = userDOB;

}

public void setUserPAN(String userPAN) {

this.userPAN = userPAN;

}

public void setUserAddress(String userAddress) {

this.userAddress = userAddress;

}

public void setNominee(String nominee) {

this.nominee = nominee;

}

public void setPhoneNumber(String phoneNumber) {

this.phoneNumber = phoneNumber;

}

public void setUserAadhar(String userAadhar) {

this.userAadhar = userAadhar;

}

}

**--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

* **OUTPUT:**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

BR14x Bank System

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1-Login User

2-Create an Account

3-Deactivate Account

4-Show Bank Accounts

5-Exit

: 2

Welcome , Please fill the details to open your account

Enter Full Name : Bhavik Ransubhe

Enter Date Of Birth : 14/01/2000

Enter Address : Salisbury Park , Pune

Enter Nominee : Girish Ransubhe

Enter PAN : EMP0969BR

Enter Phone Number : 9090909090

Enter Aadhar : 123456654321

Enter Initial Balance (Must Be Above Rs.5000) : 10000

Creating An Account.....

Please Enter New Password

Should be of minimum size 8 with Letter, Digit And Special Characters

(Remember For Next Login) : Bh@vik14

Please Enter Password Again To Confirm :Bh@vik14

Account Created With Account Number :369801

With Password :Bh@vik14

Account Created

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

BR14x Bank System

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1-Login User

2-Create an Account

3-Deactivate Account

4-Show Bank Accounts

5-Exit

: 2

Welcome , Please fill the details to open your account

Enter Full Name : Advait Thakur

Enter Date Of Birth : 12/09/2000

Enter Address : Hadapsar , Pune

Enter Nominee : Bhavik Ransubhe

Enter PAN : EMP0969AT

Enter Phone Number : 8080808080

Enter Aadhar : 111122223333

Enter Initial Balance (Must Be Above Rs.5000) : 15000

Creating An Account.....

Please Enter New Password

Should be of minimum size 8 with Letter, Digit And Special Characters

(Remember For Next Login) : Adv@it12

Please Enter Password Again To Confirm :Adv@it12

Account Created With Account Number :369802

With Password :Adv@it12

Account Created

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

BR14x Bank System

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1-Login User

2-Create an Account

3-Deactivate Account

4-Show Bank Accounts

5-Exit

: 1

Enter Account Number:369801

Enter Password :Bh@vik14

WELCOME TO BR14x SALISBURY BRANCH BANK

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Account No Name Phone Number Aadhar No PAN No DOB Address Nominee Account Balance

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

369801 Bhavik Ransubhe 9090909090 123456654321 EMP0969BR 14/01/2000 Salisbury Park , Pune Girish Ransubhe 10000.0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

| CUSTOMER PORTAL |

1-Show Account Details

2-Withdraw Amount

3-Deposit Amount

4-Transfer Amount

5-Update Account Details

6-Show Statement

7-LogOut : 2

Enter The Amount You Withdraw : 500

Amount Withdrawn

Account Balance : 9500.0

| CUSTOMER PORTAL |

1-Show Account Details

2-Withdraw Amount

3-Deposit Amount

4-Transfer Amount

5-Update Account Details

6-Show Statement

7-LogOut : 3

Enter The Amount You Deposit : 1000

Amount Deposited

Account Balance :10500.0

| CUSTOMER PORTAL |

1-Show Account Details

2-Withdraw Amount

3-Deposit Amount

4-Transfer Amount

5-Update Account Details

6-Show Statement

7-LogOut : 5

Enter Field To Change

1.Name

2.DOB

3.PAN

4.Address

5.Nominee

6.PhoneNumber

7.Aadhar No1

Current Name :Bhavik Ransubhe

Enter Change Name To :Bhavik Girish Ransubhe

Change Name [1/0]?1

| CUSTOMER PORTAL |

1-Show Account Details

2-Withdraw Amount

3-Deposit Amount

4-Transfer Amount

5-Update Account Details

6-Show Statement

7-LogOut : 1

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Account No Name Phone Number Aadhar No PAN No DOB Address Nominee Account Balance

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

369801 Bhavik Girish Ransubhe9090909090 123456654321 EMP0969BR 14/01/2000 Salisbury Park , Pune Girish Ransubhe 10500.0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

| CUSTOMER PORTAL |

1-Show Account Details

2-Withdraw Amount

3-Deposit Amount

4-Transfer Amount

5-Update Account Details

6-Show Statement

7-LogOut : 6

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date Amount Type Balance Description

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

27:10:2020 11:39 am 500.0 Withdraw 9500.0 Withdrawn Self

27:10:2020 11:40 am 1000.0 Deposit 10500.0 Deposited Self

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

| CUSTOMER PORTAL |

1-Show Account Details

2-Withdraw Amount

3-Deposit Amount

4-Transfer Amount

5-Update Account Details

6-Show Statement

7-LogOut : 7

Pass :Bh@vik14

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

BR14x Bank System

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1-Login User

2-Create an Account

3-Deactivate Account

4-Show Bank Accounts

5-Exit

: 1

Enter Account Number:369801

Enter Password :Bh@vik14

WELCOME TO BR14x SALISBURY BRANCH BANK

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Account No Name Phone Number Aadhar No PAN No DOB Address Nominee Account Balance

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

369801 Bhavik Girish Ransubhe9090909090 123456654321 EMP0969BR 14/01/2000 Salisbury Park , Pune Girish Ransubhe 10500.0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

| CUSTOMER PORTAL |

1-Show Account Details

2-Withdraw Amount

3-Deposit Amount

4-Transfer Amount

5-Update Account Details

6-Show Statement

7-LogOut : 4

TRANSFER MONEY PORTAL

Enter The Name of Account Holder :Advait Thakur

Enter The Account Number Of Holder To Transfer Amount:369802

Enter Amount to Transfer :2000

| CUSTOMER PORTAL |

1-Show Account Details

2-Withdraw Amount

3-Deposit Amount

4-Transfer Amount

5-Update Account Details

6-Show Statement

7-LogOut : 6

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date Amount Type Balance Description

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

27:10:2020 11:39 am 500.0 Withdraw 9500.0 Withdrawn Self

27:10:2020 11:40 am 1000.0 Deposit 10500.0 Deposited Self

27:10:2020 11:46 am 2000.0 Withdraw 2000.0 Transferred To Advait Thakur

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

| CUSTOMER PORTAL |

1-Show Account Details

2-Withdraw Amount

3-Deposit Amount

4-Transfer Amount

5-Update Account Details

6-Show Statement

7-LogOut : 7

Pass :Bh@vik14

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

BR14x Bank System

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1-Login User

2-Create an Account

3-Deactivate Account

4-Show Bank Accounts

5-Exit

: 4

Enter Master Password :9669

Loading Data

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Account No Name Phone Number Aadhar No PAN No DOB Address Nominee Account Balance

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

369801 Bhavik Girish Ransubhe9090909090 123456654321 EMP0969BR 14/01/2000 Salisbury Park , Pune Girish Ransubhe 8500.0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

369802 Advait Thakur 8080808080 111122223333 EMP0969AT 12/09/2000 Hadapsar , Pune Bhavik Ransubhe 17000.0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

BR14x Bank System

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1-Login User

2-Create an Account

3-Deactivate Account

4-Show Bank Accounts

5-Exit

:

--------------------------------------------------------------------------------------------------------------------------

END

--------------------------------------------------------------------------------------------------------------------------