CODE:

```
package MiniProject;

/* STATEMENT : To create a CALL CAFE using BINARY
SEARCH TREE



/* Data Structures used :
 * BST --> To store the contact details (name and
number)
 * Stack --> To store the call history
 */


/*
 * CONTRIBUTERS :
 * UCE2021616 --> Simran Desai
 * UCE2021619 --> Sanika Deshpande
 * UCE2021620 --> Gargee Dorle
 * UCE2021621 --> Mayuri Gadhave
 */



//importing library
import java.util.*;



//class to represent the binary search tree
class Node
{
	Node next;

	//declaring customer details
	String customer_name;
	String customer_number;
	Node rc;
	Node lc;
```

```
    Node(String customer_name, String
customer_number) // constructor
    {
        this.customer_name = customer_name;
        this.customer_number = customer_number;
        lc = rc = null;
    }
}



//class to represent the operations
class Function
{
    Scanner sc = new Scanner(System.in);

    //import stack from stack framework
    //to implement call function
    Stack<String> stac = new Stack<>();
    int count = 0;

    //root represents the root node of the
    //binary search tree structure
    Node root = null;



    //login into the account
    boolean login()
    {

        System.out.print("                   Password
: ");
        String password = sc.next();

        //predefined password
        String hardcode = "India@123";

        //compare passwords
        //loop till correct password is not entered
        while(password.compareTo(hardcode) != 0)
        {
            System.out.println("
Incorrect password,try again!");
```

```
            login();
            return false;
        }
        return true;
    }


    //creation of binary tree
    //storing name and number of customers
    void create()
    {
        String name = "", number = "";

        //loop till user wants to add a new contact
        do {

            System.out.print("\nEnter contact name :
");

            name = sc.next();
            name += sc.nextLine();

            //validation for the number to be a 10
digit number
            do
            {
                System.out.print("Enter contact
number : ");
                number=sc.nextLine();
                if(number.length() != 10)
                {
                    System.out.println("INVALID
NUMBER!! PLEASE ENTER VALID NUMBER");
                }
            } while(number.length() != 10) ;



            //create a new node to store
            //customer name and number
            Node newNode = new Node(name,number);
            //for the root contact
            if(root == null)
            {
                root = newNode;
            }


            //for rest of the contacts
            else
            {
                //tempory variable to percolate the
tree structure
                Node ptr = root;

                //insert contact names such that
they follow BST property
                while(ptr != null)
                {
                    //new contact added to the left

    if(ptr.customer_name.compareToIgnoreCase(newNode.
customer_name) > 0)
                    {
                        if(ptr.lc == null)
                        {
                            ptr.lc = newNode;
                            ptr = null;
                        }

                        else
                        {
                            ptr = ptr.lc;
                        }
                    }

                    //new contact added to the right
                    else
if(ptr.customer_name.compareToIgnoreCase(newNode.cust
omer_name) < 0)
                    {
                        if(ptr.rc == null)
                        {
                            ptr.rc = newNode;
                            ptr = null;
                        }

                        else
                        {
                            ptr = ptr.rc;
                        }
                    }
                }
            }


            System.out.print("\nDo you wish to add
more details?If YES press (1),if NO press (0).");
            count = sc.nextInt();
            sc.nextLine();
        }while(count == 1);
    }




    //display contacts in recursive inorder fashion
    void display(Node root)
    {
        //recursively percolate the binary search
tree
        if(root!=null)
        {

            display(root.lc);
            System.out.println(root.customer_name
+"\t\t\t\t    "+ root.customer_number);
            display(root.rc);
        }
    }



    //search for a contact according to name
    boolean Search(String name)
    {

        //base case : contact list is empty
        if(root == null)
        {
            System.out.println("\nYOUR MOBILE BOOK
IS EMPTY :(");
            return false;
        }

        int flag = 0;
        Node ptr = root;

        //percolate the tree
        while(ptr != null)
        {
            //contact found --> set flag to 1

    if(ptr.customer_name.compareToIgnoreCase(name) ==
0)
            {
                flag = 1;
                break;
            }

            //else search in the left side
            else
if(ptr.customer_name.compareToIgnoreCase(name) > 0)
            {
                ptr = ptr.lc;
            }

            //else search in the right side
            else
if(ptr.customer_name.compareToIgnoreCase(name) < 0)
            {
                ptr = ptr.rc;
            }
        }

        //flag conditions
        if(flag == 1)
        {
            System.out.println("Contact found : ");
            System.out.println("CONTACT NAME\t\t
CONTACT NUMBER");
            System.out.print(ptr.customer_name +
"\t\t\t" + ptr.customer_number);
```

```
            return true;
        }

        //validation : contact not saved
        else if(flag==0)
        {
            System.out.println("NUMBER NOT
FOUND!!!");
        }

        return false;
    }



    //update name or number of the contact
    void update()
    {

        //base case : their are no contacts in the
phone book
        if(root == null)
        {
            System.out.println("\nMOBILE BOOK IS
EMPTY :(");
            return;
        }

        int flag = 0;

        System.out.println("Enter name");
        String name = sc.next();
        name += sc.nextLine();

        System.out.println("0. Exit \n1.Update name :
\n2.Update number : ");
        System.out.println("Enter the your choice :
");
        int choice = sc.nextInt();

        //switch case for updating name or number
        switch(choice)
        {
        //exit the feature
        case 0:
            System.exit(0);
            break;

            //update contact name
        case 1:
            Node temp = root;
            if(Search(name) == true)
            {
                //then ask for the update they want
to make
                System.out.println();
                System.out.println("\nEnter the
updated name : ");
                String newName = sc.next();
                newName += sc.nextLine();

                //searching for the contact
                while(temp != null)
                {

    if(temp.customer_name.compareToIgnoreCase(name)
== 0)
                    {
                        //contact found and
                        //value is updated
                        temp.customer_name =
newName;

                        flag = 1;
                        break;
                    }

                    else
if(temp.customer_name.compareToIgnoreCase(name) > 0)
                    {
                        temp = temp.lc;
                    }

                    else
if(temp.customer_name.compareToIgnoreCase(name) < 0)
                    {
                        temp = temp.rc;
                    }
                }
            }

            //validation :
            //if contact not found
            if(flag==0)
            {
                System.out.println("SORRY!CONTACT
NOT FOUND :(");
                break;

            }

            break;


            //update number of a contact
        case 2 :
            //input the name to be updated
            Node tem = root;
            String newNumber ="";
            //flag
            int h = 0;

            if(Search(name) == true)
            {
                //then ask for the update they want
to make
                do
                {
                    System.out.println("Enter the
updated number : ");
                    newNumber = sc.next();
                    if(newNumber.length() != 10)
                    {
                        System.out.println("INVALID
NUMBER!! PLEASE ENTER VALID NUMBER");
                    }
                } while(newNumber.length() != 10) ;


                while(tem != null)
                {
```

```
    if(tem.customer_name.compareToIgnoreCase(name) ==
0)
                    {
                        h++;
                        //update the number
                        tem.customer_number =
newNumber;
                        break;
                    }

                    else
if(tem.customer_name.compareToIgnoreCase(name) > 0)
                    {
                        tem = tem.lc;
                    }
                    else
if(tem.customer_name.compareToIgnoreCase(name) < 0)
                    {
                        tem = tem.rc;
                    }
                }
            }

            //validation for contact not found
            if(h == 0)
            {
                System.out.println("SORRY!CONTACT
NOT FOUND :(");
                break;
            }
            break;


        default:
            System.out.println("Invalid choice");
            break;
        }
    }




    //recursively call the delete function
    void delete(String del)
    {
        int flag = 0;
        if(Search(del))
        {
            root = deleterec(root, del);
            flag = 1;
        }


        if(flag == 1)
        {
            System.out.println("\nCONTACT DELETED
SUCCESSFULLY :)");
        }
    }


    //delete contact from the phone book
    Node deleterec(Node root,String del) {

        //base case : their are no contacts in the
phone book
        if(root == null)
        {
            return root;
        }

        //search for the contact based on name first

    if(root.customer_name.compareToIgnoreCase(del) >
0)
        {
            root.lc = deleterec(root.lc,del);
        }

        else
if(root.customer_name.compareToIgnoreCase(del) < 0)
        {
            root.rc = deleterec(root.rc,del);
        }

        //contact found
        else
```

```
          {
               if(root.lc == null)
                    return root.rc;
               else if(root.rc == null)
                    return root.lc;

               //replace with predessor
               root.customer_name = minValue(root.rc);

               //pass the contact for delete
               root.rc = deleterec(root.rc, 
root.customer_name);
          }
          return root;
     }



     //search for the predecessor of the number to be 
deleted
     String minValue(Node ptr)
     {
          //predecessor is the right most node
          //in the left tree
          String minv = ptr.customer_name;
          while (ptr.lc != null) {
               minv = ptr.lc.customer_name;
               ptr = root.lc;
          }
          return minv;
     }



     //make a call
     void call()
     {
          String c;

          if(root == null)
          {
               System.out.println("\nYOUR MOBILE BOOK 
IS EMPTY :(");
```

```
            //to call a saved number
            do
            {
                System.out.print("Enter contact 
number:");
                c = sc.next();
                if(c.length()!= 10)
                {
                    System.out.println("INVALID 
NUMBER!! PLEASE ENTER VALID NUMBER");
                }
            }while(c.length() != 10);

            //contact found and is on call
            System.out.println("\n*********ON 
CALL***********");
            //push contact name into history stack
            stac.push(c);
            System.out.println(c);
            return;
        }

        System.out.println("Enter the name you wish 
to call");
        c = sc.next();
        c += sc.nextLine();

        //validation : contact not found
        if(!Search(c))
        {
            //to call an unsaved number
            System.out.println("CONTACT NOT FOUND 
:(");

            System.out.println("Enter number:");
            c = sc.next();

        }

        //contact found and is on call
        System.out.println("\n*********ON 
CALL***********");
        //push contact name into history stack
        stac.push(c);
        System.out.println(c);
```

```
    }


    //call history
    void History()
    {


    System.out.println("\n*******HISTORY*******");
        Stack<String> stac2 = new Stack<>();

        //validation for no calls
        if(stac.isEmpty())
        {
            System.out.println("NO CALLS YET");
        }

        //empty the stack
        else
        {
            while(!stac.isEmpty())
            {
                String histo = stac.pop();
                System.out.println(histo + " ");
                stac2.push(histo);
            }

            //push all the history information back
into the original stack
            while(!stac2.isEmpty())
            {
                String histo2 = stac2.pop();
                stac.push(histo2);
            }
        }
    }
}





//main class
```

```
public class MobileBook
{
    public static void main(String args[])
    {


         Scanner sc = new Scanner(System.in);

         //object of class to instantiate the
functions
         Function fun = new Function();
         String name = "";
         int cont = 0;


         System.out.println("\t616,619,620,621 present
before you the data structures miniproject!!");
         System.out.println("\n
LOG IN FOR CALL CAFE                   ");

         fun.login();

         System.out.println("-------------------------
---------------------------------------------------
");

    System.out.println("*************************WELC
OME TO CALL CAFE******************************");
         System.out.println("-------------------------
---------------------------------------------------
");


         do
         {
             do
             {

    System.out.println("_____________________________
______________________________________________");

    System.out.println("\n~~~~~~~~MeNu~~~~~~~~");
                 System.out.println(
                          " \n1)||ADD DETAILS||"
```

```
                                  + "
\n2)||DISPLAY DETAILS||"
                                  + "
\n3)||SEARCH A PERSON||"
                                  + "
\n4)||UPDATE DETAILS||"
                                  + "
\n5)||DELETE CONTACT||"
                                  + "
\n6)||CALL A CONTACT||"
                                  + "
\n7)||DISPLAY HISTORY||");
                 System.out.print("\nPlease enter
your choice:");
                 int ch = sc.nextInt();


                 switch(ch)
                 {

                 //to terminate the call cafe


                 //add contacts
                 case 1:
                     System.out.println("\nYou have
selected >>>> ||ADD DETAILS||");
                     fun.create();
// Call create function
                     break;


                     //display the contact details
                 case 2:
                     System.out.println("\nYou have
selected >>>> ||DISPLAY DETAILS||");

                     //validation
                     if(fun.root == null)
                     {
                         System.out.println("\nYOUR
MOBILE BOOK IS EMPTY :(");
                         break;
                     }
```

```
                      System.out.println("*Contact
Name*"+"\t\t\t\t"+"*Contact Number*");
                      fun.display(fun.root);
// Call display function
                      break;


                      //search for a contact
                 case 3:
                      System.out.println("\nYou have
selected >>>> ||SEARCH A PERSON||");

                      //validation
                      if(fun.root == null)
                      {
                          System.out.println("\nYOUR
MOBILE BOOK IS EMPTY :(");
                          break;
                      }

                      System.out.println("Enter name
for which you want search number");   //Call Search
function
                      sc.nextLine();
                      name = sc.nextLine();
                      fun.Search(name);
                      break;


                      //update contact name or number
                 case 4:
                      System.out.println("\nYou have
selected >>>> ||UPDATE CONTACT||");
                      fun.update();
                      break;


                      //delete a contact
                 case 5:
                      System.out.println("\nYou have
selected >>>> ||DELETE CONTACT||");

                      //validation
                      if(fun.root == null)
```

```
                    {
                        System.out.println("\nYOUR
MOBILE BOOK IS EMPTY :(");
                        break;
                    }
                    System.out.println("Enter name
to be deleted:");
                    String del = sc.next();
                    del += sc.nextLine();
                    fun.delete(del);
                    break;


                    //call a contact
                case 6:
                    System.out.println("\nYou have
selected >>>> ||CALL A CONTACT||");
                    fun.call();
                    break;


                    //history
                case 7:
                    System.out.println("\nYou have
selected >>>> ||DISPLAY HISTORY||");
                    fun.History();
                    break;


                default :
                    System.out.println("Invalid
choice");
                }
                System.out.println("\nIf you want to
continue,press '1' or else press '0' for exiting the
app.");
                cont = sc.nextInt();

            } while(cont == 1);
            System.out.println("Are you sure you
want to logout :  YES(0)  NO(1)");
            cont = sc.nextInt();

        } while(cont == 1);
        if(cont == 0)
        {
             System.out.println("Thank you for 
operating with us!!");
        }
    }
}
```

OUTPUT:

```
/*COMMENTED OUTPUT FOR THE PROGRAM
616,619,620,621 present before you the data 
structures miniproject!!

LOG IN FOR CALL CAFE
Password : INDIA@123
Incorrect password,try again!
Password : India@123
-----------------------------------------------------
-----------------------
*************************WELCOME TO CALL 
CAFE*****************************
-----------------------------------------------------
-----------------------
_____________________________________________________
______________________
```

```
~~~~~~~~MeNu~~~~~~~~

1)||ADD DETAILS||
2)||DISPLAY DETAILS||
3)||SEARCH A PERSON||
4)||UPDATE DETAILS||
5)||DELETE CONTACT||
6)||CALL A CONTACT||
7)||DISPLAY HISTORY||

Please enter your choice:2

You have selected >>>> ||DISPLAY DETAILS||

YOUR MOBILE BOOK IS EMPTY :(

If you want to continue,press '1' or else press '0'
for exiting the app.
1
_____________________________________________________
______________________

~~~~~~~~MeNu~~~~~~~~

1)||ADD DETAILS||
2)||DISPLAY DETAILS||
3)||SEARCH A PERSON||
4)||UPDATE DETAILS||
5)||DELETE CONTACT||
6)||CALL A CONTACT||
7)||DISPLAY HISTORY||

Please enter your choice:1

You have selected >>>> ||ADD DETAILS||

Enter contact name : Simran
Enter contact number : 9874563214

Do you wish to add more details?If YES press (1),if
NO press (0).1

Enter contact name : Sanika
```

```
Enter contact number : 9855656541

Do you wish to add more details?If YES press (1),if
NO press (0).1

Enter contact name : Gargee
Enter contact number : 9856325633

Do you wish to add more details?If YES press (1),if
NO press (0).1

Enter contact name : Mayuri
Enter contact number : 9999656325

Do you wish to add more details?If YES press (1),if
NO press (0).1

Enter contact name : Alfa De
Enter contact number : 4585858
INVALID NUMBER!! PLEASE ENTER VALID NUMBER
Enter contact number : 9856325236

Do you wish to add more details?If YES press (1),if
NO press (0).0

If you want to continue,press '1' or else press '0'
for exiting the app.
1
_____________________________________________________
______________________

~~~~~~~~MeNu~~~~~~~~

1)||ADD DETAILS||
2)||DISPLAY DETAILS||
3)||SEARCH A PERSON||
4)||UPDATE DETAILS||
5)||DELETE CONTACT||
6)||CALL A CONTACT||
7)||DISPLAY HISTORY||

Please enter your choice:2

You have selected >>>> ||DISPLAY DETAILS||
```

```
*Contact Name*                *Contact Number*
Alfa De                   9856325236
Gargee                    9856325633
Mayuri                    9999656325
Sanika                    9855656541
Simran                    9874563214

If you want to continue,press '1' or else press '0'
for exiting the app.
1
_____________________________________________________
______________________

~~~~~~~~MeNu~~~~~~~~

1)||ADD DETAILS||
2)||DISPLAY DETAILS||
3)||SEARCH A PERSON||
4)||UPDATE DETAILS||
5)||DELETE CONTACT||
6)||CALL A CONTACT||
7)||DISPLAY HISTORY||

Please enter your choice:3

You have selected >>>> ||SEARCH A PERSON||
Enter name for which you want search number
Riya
NUMBER NOT FOUND!!!

If you want to continue,press '1' or else press '0'
for exiting the app.
1
_____________________________________________________
______________________

~~~~~~~~MeNu~~~~~~~~

1)||ADD DETAILS||
2)||DISPLAY DETAILS||
3)||SEARCH A PERSON||
4)||UPDATE DETAILS||
5)||DELETE CONTACT||
6)||CALL A CONTACT||
```

```
7)||DISPLAY HISTORY||

Please enter your choice:3

You have selected >>>> ||SEARCH A PERSON||
Enter name for which you want search number
Simran
Contact found :
CONTACT NAME      CONTACT NUMBER
Simran           9874563214
If you want to continue,press '1' or else press '0'
for exiting the app.
1
_____________________________________________________
______________________

~~~~~~~~MeNu~~~~~~~~

1)||ADD DETAILS||
2)||DISPLAY DETAILS||
3)||SEARCH A PERSON||
4)||UPDATE DETAILS||
5)||DELETE CONTACT||
6)||CALL A CONTACT||
7)||DISPLAY HISTORY||

Please enter your choice:4

You have selected >>>> ||UPDATE CONTACT||
Enter name
Gargee
0. Exit
1.Update name :
2.Update number :
Enter the your choice :
1
Contact found :
CONTACT NAME      CONTACT NUMBER
Gargee           9856325633

Enter the updated name :
Gargi
```

```
If you want to continue,press '1' or else press '0'
for exiting the app.
1
_____________________________________________________
______________________

~~~~~~~~MeNu~~~~~~~~

1)||ADD DETAILS||
2)||DISPLAY DETAILS||
3)||SEARCH A PERSON||
4)||UPDATE DETAILS||
5)||DELETE CONTACT||
6)||CALL A CONTACT||
7)||DISPLAY HISTORY||

Please enter your choice:4

You have selected >>>> ||UPDATE CONTACT||
Enter name
Gargee
0. Exit
1.Update name :
2.Update number :
Enter the your choice :
2
NUMBER NOT FOUND!!!
SORRY!CONTACT NOT FOUND :(

If you want to continue,press '1' or else press '0'
for exiting the app.
1
_____________________________________________________
______________________

~~~~~~~~MeNu~~~~~~~~

1)||ADD DETAILS||
2)||DISPLAY DETAILS||
3)||SEARCH A PERSON||
4)||UPDATE DETAILS||
5)||DELETE CONTACT||
6)||CALL A CONTACT||
7)||DISPLAY HISTORY||
```

```
Please enter your choice:4

You have selected >>>> ||UPDATE CONTACT||
Enter name
Gargi
0. Exit
1.Update name :
2.Update number :
Enter the your choice :
2
Contact found :
CONTACT NAME      CONTACT NUMBER
Gargi            9856325633Enter the updated number :
7485965874

If you want to continue,press '1' or else press '0'
for exiting the app.
1
_____________________________________________________
______________________

~~~~~~~~MeNu~~~~~~~~

1)||ADD DETAILS||
2)||DISPLAY DETAILS||
3)||SEARCH A PERSON||
4)||UPDATE DETAILS||
5)||DELETE CONTACT||
6)||CALL A CONTACT||
7)||DISPLAY HISTORY||

Please enter your choice:5

You have selected >>>> ||DELETE CONTACT||
Enter name to be deleted:
Sanika
Contact found :
CONTACT NAME      CONTACT NUMBER
Sanika           9855656541
CONTACT DELETED SUCCESSFULLY :)

If you want to continue,press '1' or else press '0'
for exiting the app.
```

```
1
______________________________________________________
______________________

~~~~~~~~MeNu~~~~~~~~

1)||ADD DETAILS||
2)||DISPLAY DETAILS||
3)||SEARCH A PERSON||
4)||UPDATE DETAILS||
5)||DELETE CONTACT||
6)||CALL A CONTACT||
7)||DISPLAY HISTORY||

Please enter your choice:2

You have selected >>>> ||DISPLAY DETAILS||
*Contact Name*                 *Contact Number*
Alfa De                   9856325236
Gargi                     7485965874
Mayuri                    9999656325
Simran                    9874563214

If you want to continue,press '1' or else press '0'
for exiting the app.
1
______________________________________________________
______________________

~~~~~~~~MeNu~~~~~~~~

1)||ADD DETAILS||
2)||DISPLAY DETAILS||
3)||SEARCH A PERSON||
4)||UPDATE DETAILS||
5)||DELETE CONTACT||
6)||CALL A CONTACT||
7)||DISPLAY HISTORY||

Please enter your choice:6

You have selected >>>> ||CALL A CONTACT||
Enter the name you wish to call
Mayuri
```

```
Contact found :
CONTACT NAME      CONTACT NUMBER
Mayuri           9999656325
*********ON CALL***********
Mayuri

If you want to continue,press '1' or else press '0'
for exiting the app.
1
_____________________________________________________
______________________

~~~~~~~~MeNu~~~~~~~~

1)||ADD DETAILS||
2)||DISPLAY DETAILS||
3)||SEARCH A PERSON||
4)||UPDATE DETAILS||
5)||DELETE CONTACT||
6)||CALL A CONTACT||
7)||DISPLAY HISTORY||

Please enter your choice:6

You have selected >>>> ||CALL A CONTACT||
Enter the name you wish to call
Riya
NUMBER NOT FOUND!!!
CONTACT NOT FOUND :(
Enter number:
8149460125

*********ON CALL***********
8149460125

If you want to continue,press '1' or else press '0'
for exiting the app.
1
_____________________________________________________
______________________

~~~~~~~~MeNu~~~~~~~~

1)||ADD DETAILS||
```

```
2)||DISPLAY DETAILS||
3)||SEARCH A PERSON||
4)||UPDATE DETAILS||
5)||DELETE CONTACT||
6)||CALL A CONTACT||
7)||DISPLAY HISTORY||

Please enter your choice:6

You have selected >>>> ||CALL A CONTACT||
Enter the name you wish to call
Simran
Contact found :
CONTACT NAME      CONTACT NUMBER
Simran           9874563214
*********ON CALL***********
Simran

If you want to continue,press '1' or else press '0'
for exiting the app.
1
_____________________________________________________
______________________

~~~~~~~~MeNu~~~~~~~~

1)||ADD DETAILS||
2)||DISPLAY DETAILS||
3)||SEARCH A PERSON||
4)||UPDATE DETAILS||
5)||DELETE CONTACT||
6)||CALL A CONTACT||
7)||DISPLAY HISTORY||

Please enter your choice:7

You have selected >>>> ||DISPLAY HISTORY||

*******HISTORY*******
Simran
8149460125
Mayuri
```

```
If you want to continue,press '1' or else press '0'
for exiting the app.
0
Are you sure you want to logout :  YES(0)  NO(1)
1
_____________________________________________________
______________________

~~~~~~~~MeNu~~~~~~~~

1)||ADD DETAILS||
2)||DISPLAY DETAILS||
3)||SEARCH A PERSON||
4)||UPDATE DETAILS||
5)||DELETE CONTACT||
6)||CALL A CONTACT||
7)||DISPLAY HISTORY||

Please enter your choice:7

You have selected >>>> ||DISPLAY HISTORY||

*******HISTORY*******
Simran
8149460125
Mayuri

If you want to continue,press '1' or else press '0'
for exiting the app.
0
Are you sure you want to logout :  YES(0)  NO(1)
0
Thank you for operating with us!!
*/
```