**T E L E P H O N E M A N A G E M E N T S Y S T E M**

This is a JAVA program to store the names and telephone numbers in a *Telephone Management System.* The user is given a menu of 5 commands:

1. Add a new phone number
2. Display the contact list
3. Search a contact
4. Delete a contact
5. Exit Program

*Coding:*

import java.io.\*;

public class Telephone\_Numbers

{

public static void main (String[] args) throws IOException

{

BufferedReader br = new BufferedReader (new InputStreamReader (System.in));

System.out.println ();

System.out.println ("\t\t\*\*\*\*\*\*\*\*\*\* Welcome to Telephone Management System \*\*\*\*\*\*\*\*\*\*");

System.out.println ();

int capacity = 0;

int count = 0;

do

{

System.out.print ("Enter the capacity of your phone upto which it can store contacts: ");

capacity = Integer.parseInt (br.readLine ());

} while ((capacity < 0) && System.out.printf ("Invalid!!! Capacity of your phone cannot be less than 1.") != null);

String[] name = new String [capacity];

String[] phno = new String [capacity];

String[] arr = new String [capacity];

int ch = 0;

System.out.println ("\f");

int really\_Exit = 'N';

while (ch < 5)

{

while (really\_Exit == 'N')

{

System.out.println ();

System.out.println ("\t\t \*\*\*\*\*\*\*\*\*\* MAIN MENU \*\*\*\*\*\*\*\*\*\*");

System.out.println ("\t 1. Add a new phone number");

System.out.println ("\t 2. Display the contact list");

System.out.println ("\t 3. Search a phone number");

System.out.println ("\t 4. Delete a phone number");

System.out.println ("\t 5. Exit application");

System.out.println ();

do

{

System.out.println ("Enter your choice (1-5): ");

ch = Integer.parseInt (br.readLine ());

} while ((ch < 1 || ch > 5) && System.out.printf ("%s%n", "Invalid!!! Choice should be in between (1-5)") != null);

switch (ch)

{

case 1:

System.out.println ("You have opted for option " + ch + ".");

System.out.println ();

char ans = ' ';

do

{

System.out.print ("Would you really like to add a phone number? (y/n, where y=YES & n=NO): ");

ans = br.readLine (). charAt (0);

ans = Character.toUpperCase (ans);

} while (("YN".indexOf(ans) < 0) && System.out.printf ("%s%n", "Invalid Choice!!! Must be y or n") != null);

System.out.println ();

if (ans == 'Y')

{

for (int i = 0 ; i < 1 ; i++)

{

for (int j = count ; j < (count+1) ; j++)

{

System.out.println ("Enter the name: ");

name[j] = br.readLine();

System.out.println ("Enter the phone number: ");

phno[j] = br.readLine();

}

count++;

}

System.out.println ();

System.out.println ();

System.out.println ("Saving contact...");

for (int i = 0 ; i <= 1000000000 ; i++)

{

}

System.out.println ();

System.out.println ("New contact has been successfully saved.");

for (int i = 0 ; i <= 1000000000 ; i++)

{

}

System.out.println ();

System.out.println ();

}

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

{

for (int j = 0 ; j < count - 1 -i ; j++)

{

if (name[j].compareTo (name[j+1]) >0)

{

String temp = name[j];

name[j] = name[j+1];

name[j+1] = temp;

}

}

}

System.out.println ("Press <ENTER> to proceed.");

br.readLine();

System.out.println ("\f");

break;

case 2:

System.out.println ("You have opted for option " + ch + ".");

System.out.println ();

ans = ' ';

if (name[0] == null)

{

System.out.println ("No contact exists.");

}

else if (name[0] != null)

{

do

{

System.out.print ("Would you really like to display the contact list? (y/n, where y=YES & n=NO): ");

ans = br.readLine (). charAt (0);

ans = Character.toUpperCase (ans);

} while (("YN".indexOf(ans) < 0) && System.out.printf ("%s%n", "Invalid Choice!!! Must be y or n") != null);

System.out.println ();

if (ans == 'Y')

{

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

{

if (!name[i].equals (arr[i]))

{

System.out.println ("Name: " + name[i]);

System.out.println ("Phone Number: " + phno[i]);

System.out.println ();

}

}

}

}

System.out.println ("Press <ENTER> to proceed.");

br.readLine();

System.out.println ("\f");

break;

case 3:

System.out.println ("You have opted for option " + ch + ".");

System.out.println ();

ans = ' ';

if (name[0] == null)

{

System.out.println ("No contact exists.");

}

else if (name[0] != null)

{

do

{

System.out.print ("Would you really like to search for a contact? (y/n, where y=YES & n=NO): ");

ans = br.readLine (). charAt (0);

ans = Character.toUpperCase (ans);

} while (("YN".indexOf(ans) < 0) && System.out.printf ("%s%n", "Invalid Choice!!! Must be y or n") != null);

System.out.println ();

System.out.println ("The already existing contacts are: ");

System.out.println ();

if (ans == 'Y')

{

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

{

if (!name[i].equals (arr[i]))

{

System.out.println ("Name: " + name[i]);

System.out.println ("Phone Number: " + phno[i]);

System.out.println ();

}

}

System.out.println ("Enter the full name or the beginning character(s) of the contact name that is to searched: ");

String search = br.readLine ().toLowerCase ();

int found = 0;

System.out.println ();

for (int i = 0 ; i <= 1000000000 ; i++)

{

}

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

{

String s = name[i].toLowerCase ();

if (s.startsWith (search))

{

System.out.println ();

System.out.println ("Name: " + name[i]);

System.out.println ("Phone Number: " + phno[i]);

System.out.println ();

found = 1;

}

}

if (found == 0)

{

System.out.println ("No contact with such specifications exists.");

System.out.println ();

}

}

}

System.out.println ("Press <ENTER> to proceed.");

br.readLine();

System.out.println ("\f");

break;

case 4:

System.out.println ("You have opted for option " + ch + ".");

System.out.println ();

ans = ' ';

if (name[0] == null)

{

System.out.println ("No contact exists.");

}

else if (name[0] != null)

{

do

{

System.out.print ("Would you really like to delete a contact? (y/n, where y=YES & n=NO): ");

ans = br.readLine (). charAt (0);

ans = Character.toUpperCase (ans);

} while (("YN".indexOf(ans) < 0) && System.out.printf ("%s%n", "Invalid Choice!!! Must be y or n") != null);

System.out.println ();

System.out.println ("The already existing contacts are: ");

System.out.println ();

if (ans == 'Y')

{

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

{

if (!name[i].equals (arr[i]))

{

System.out.println ("Name: " + name[i]);

System.out.println ("Phone Number: " + phno[i]);

System.out.println ();

}

}

System.out.println ("Enter the full contact name that is to be deleted: ");

String delete = br.readLine ();

int foun = 0;

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

{

if (name[i].equalsIgnoreCase (delete))

{

name[i] = "";

arr[i] = name[i];

foun = 1;

System.out.println ();

System.out.println ("Deleting contact...");

for (int j = 0 ; j <= 1000000000 ; j++)

{

}

System.out.println ();

System.out.println ("Specified contact has been successfully deleted.");

for (int k = 0 ; k <= 1000000000 ; k++)

{

}

System.out.println ();

}

}

if (foun == 0)

{

System.out.println ("No contact with such specification exists.");

System.out.println ();

}

}

}

System.out.println ("Press <ENTER> to proceed.");

br.readLine();

System.out.println ("\f");

break;

case 5:

System.out.println ("You have opted for option " + ch + ".");

System.out.println ();

char really\_exit = ' ';

do

{

System.out.print ("Would you really like exit? (y/n, where y=YES & n=NO): ");

really\_exit = br.readLine (). charAt (0);

really\_exit = Character.toUpperCase (really\_exit);

} while (("YN".indexOf(really\_exit) < 0) && System.out.printf ("%s%n", "Invalid Choice!!! Must be y or n") != null);

System.out.println ();

if (really\_exit == 'Y')

{

System.out.println ("Press <ENTER> to proceed.");

br.readLine ();

System.out.println ("Thank you for using the application. Your application will soon be terminated...");

System.out.println ();

System.exit (-1);

}

System.out.println ("Press <ENTER> to proceed.");

br.readLine ();

System.out.println ("\f");

break;

}

}

}

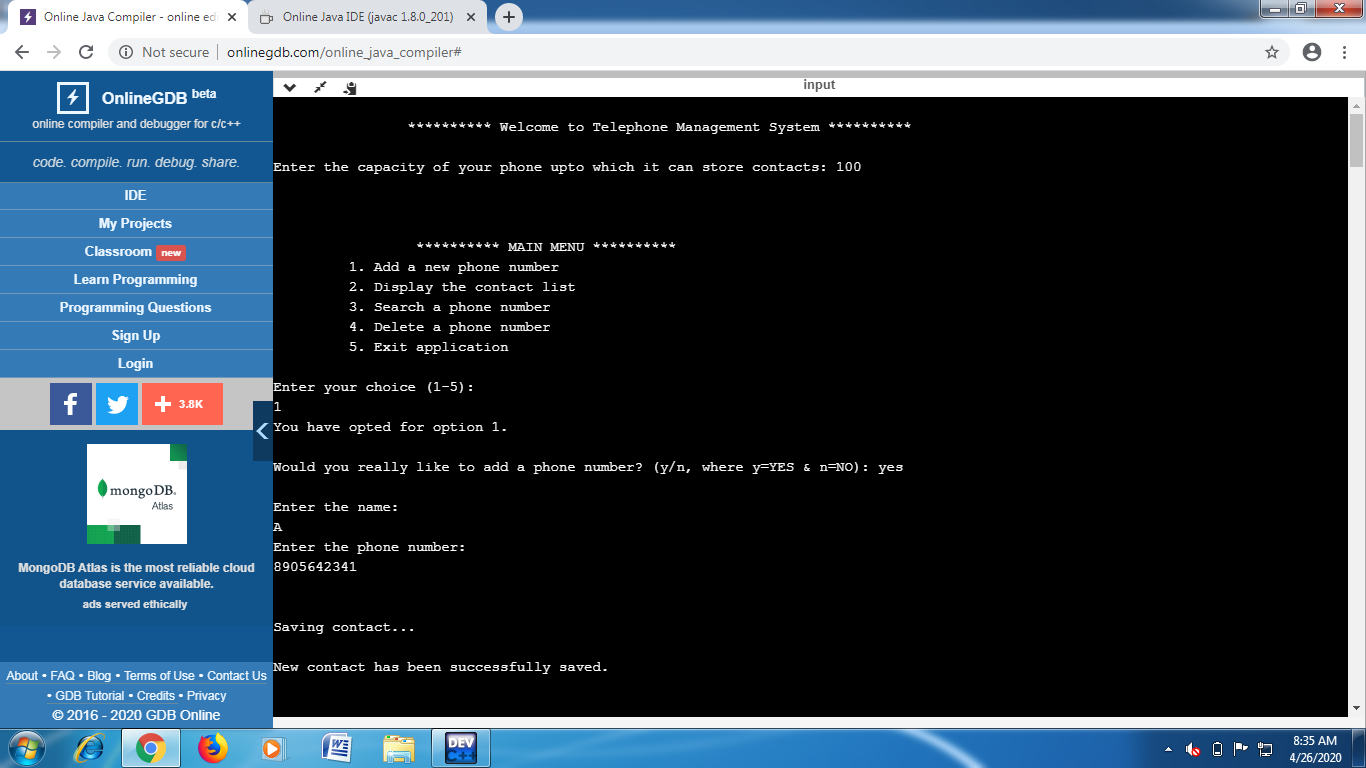
}

}

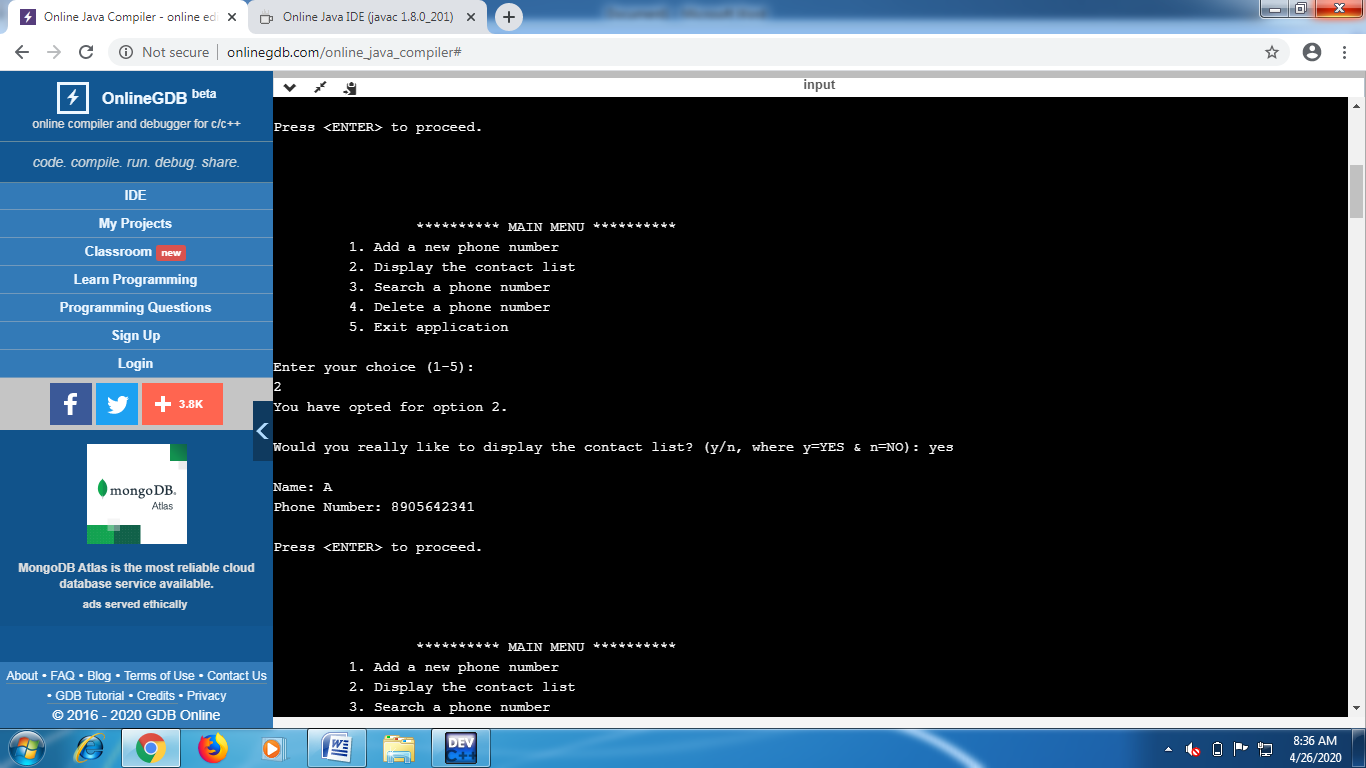
\**For OUTPUT of the program, please turn over to the next page*

*P.T.O.*

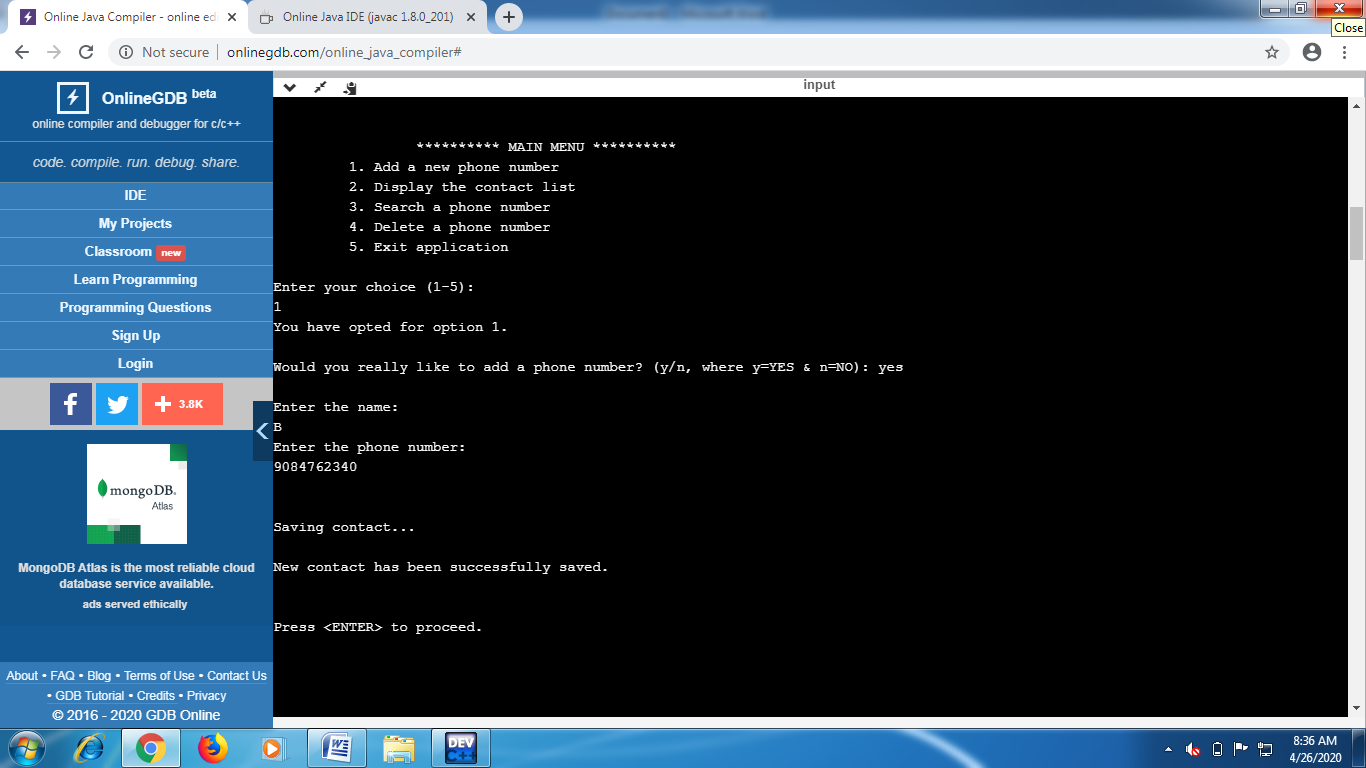
*OUTPUT*



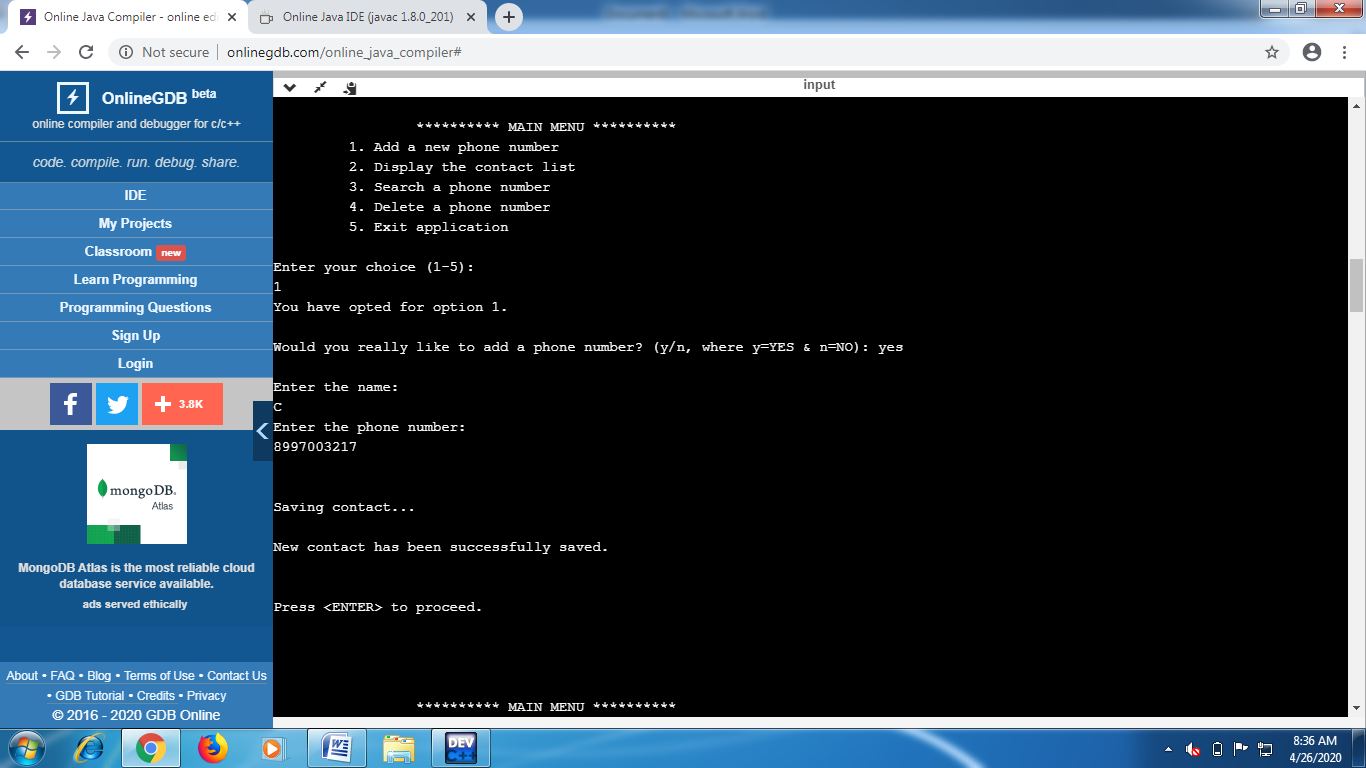
**Fig. 1:** Adding the first Contact



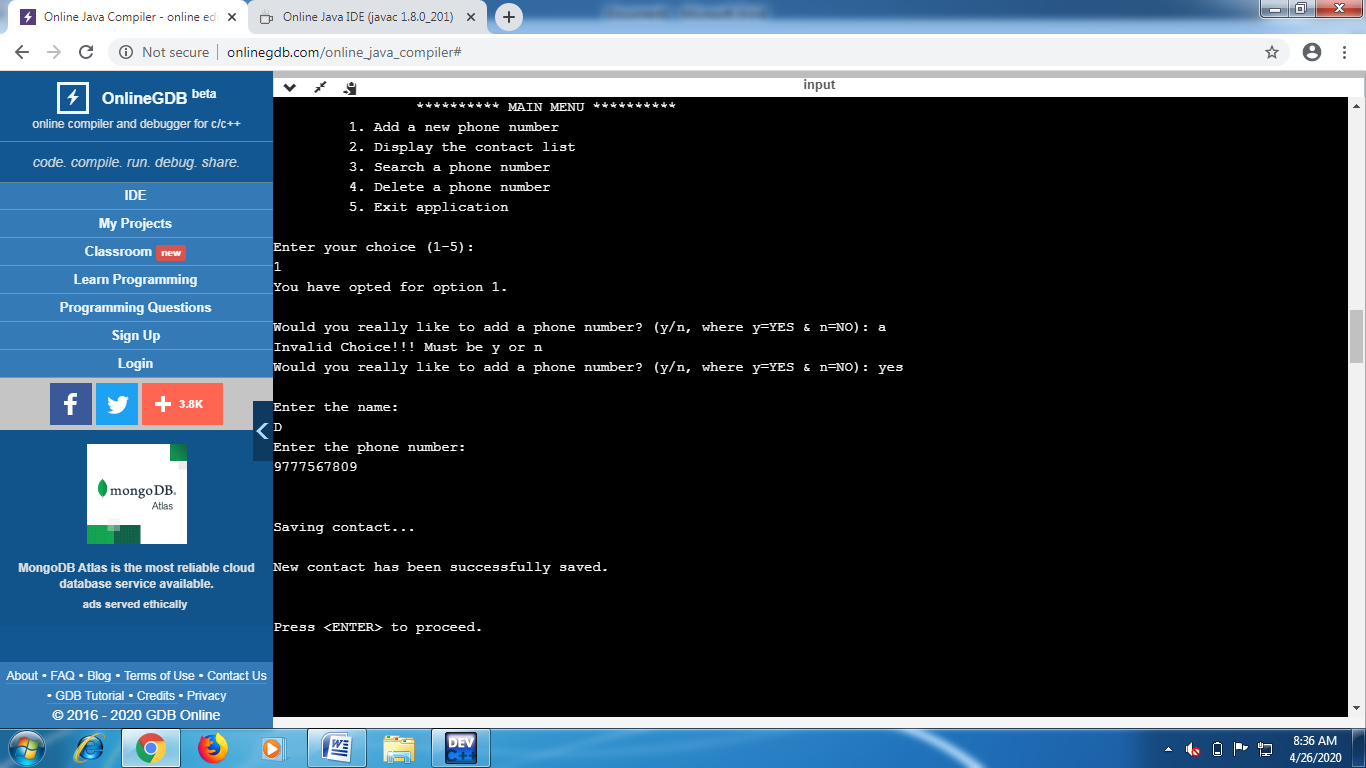
**Fig. 2:** Displaying the Contact List after adding the first Contact



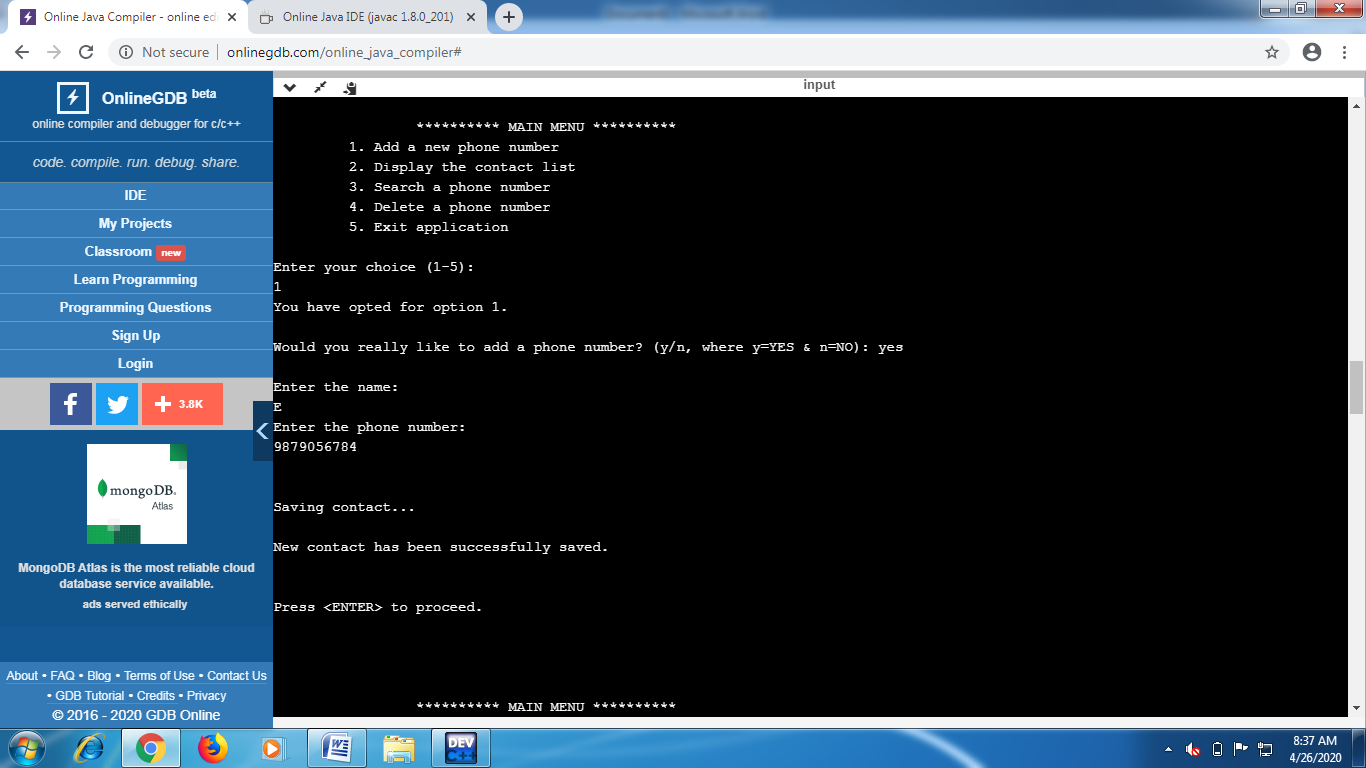
**Fig. 3:** Adding Contacts



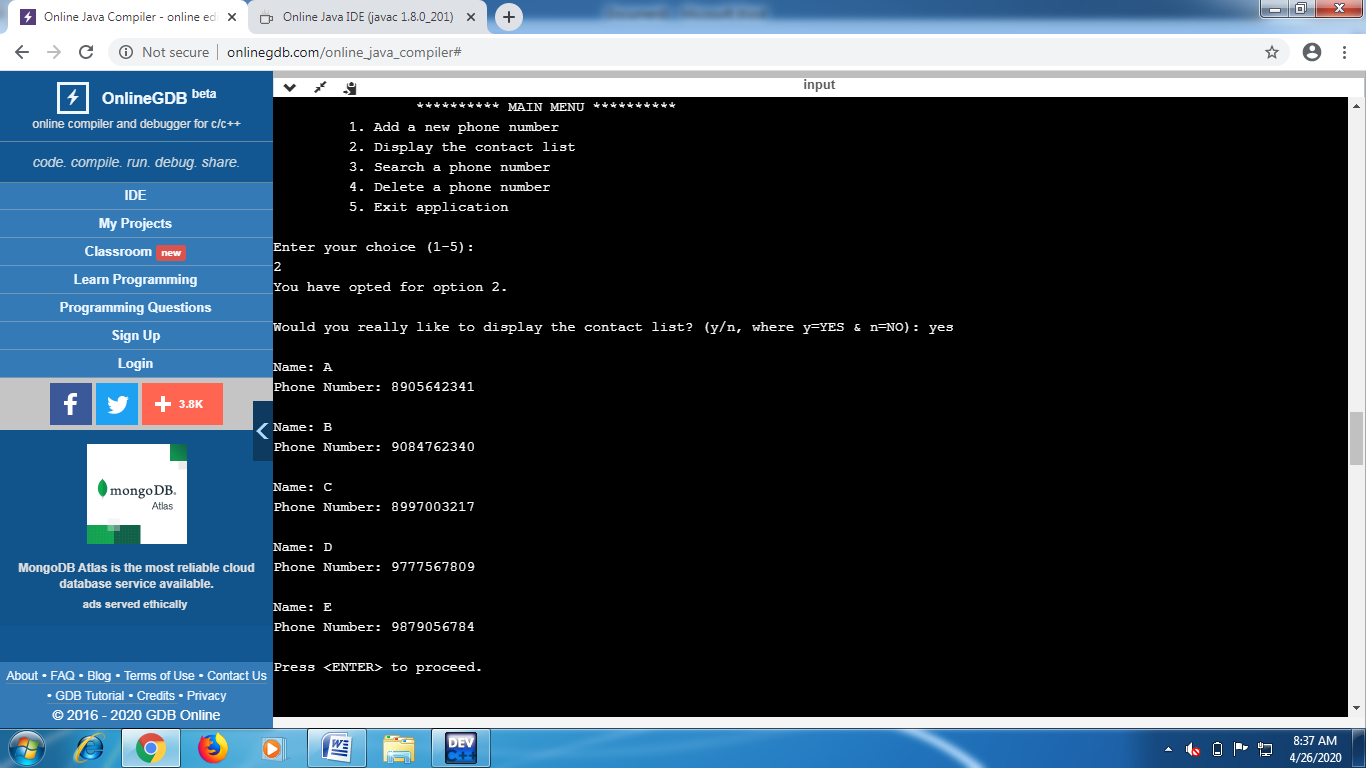
**Fig. 4:** Adding Contacts (contd…)



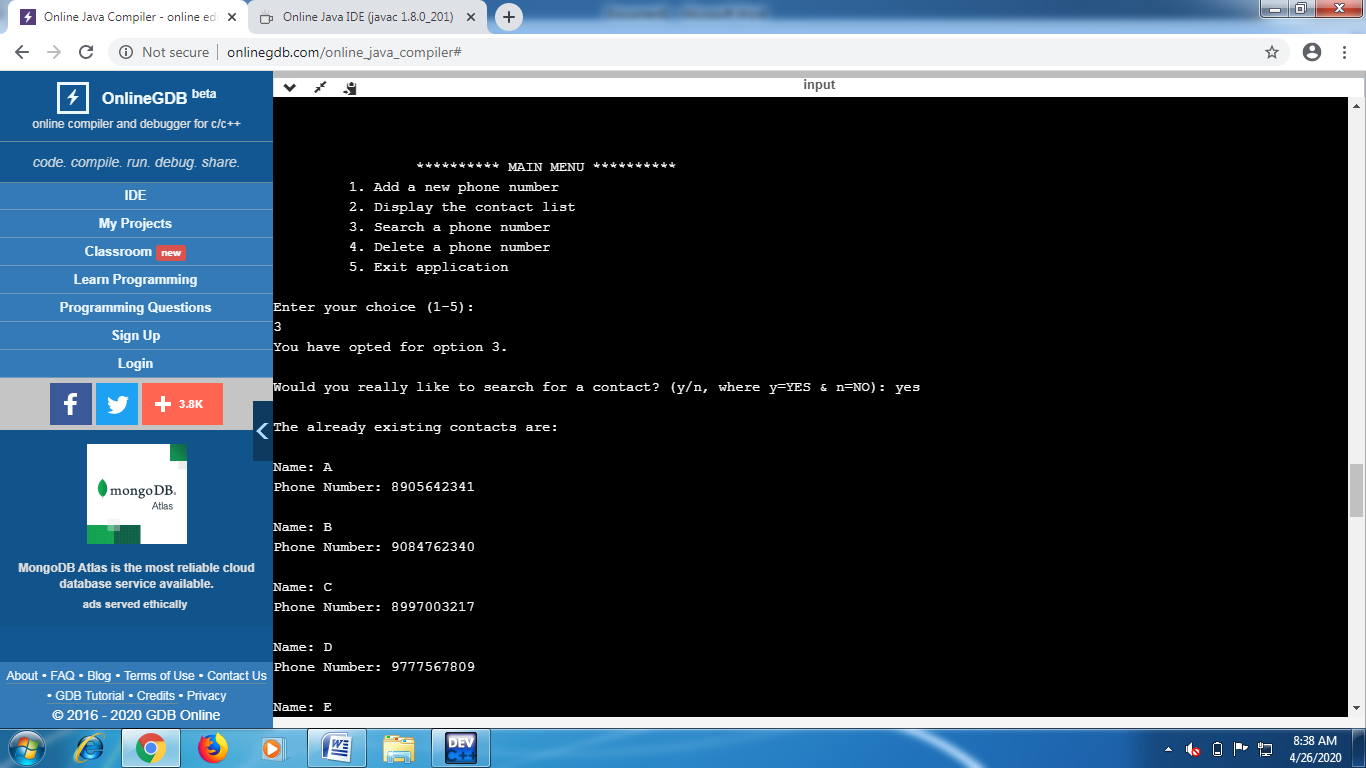
**Fig. 5:** Adding Contacts (contd…)



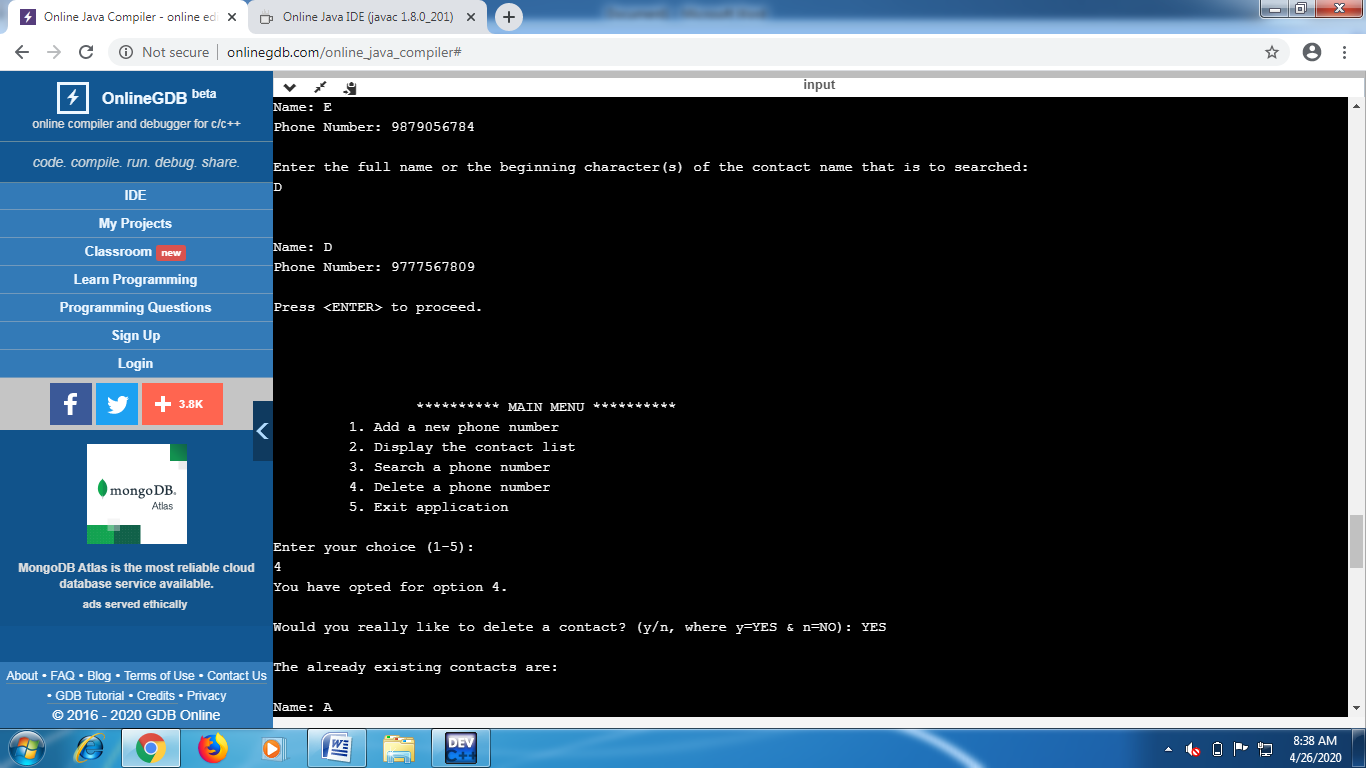
**Fig. 6:** Adding Contacts (contd…)



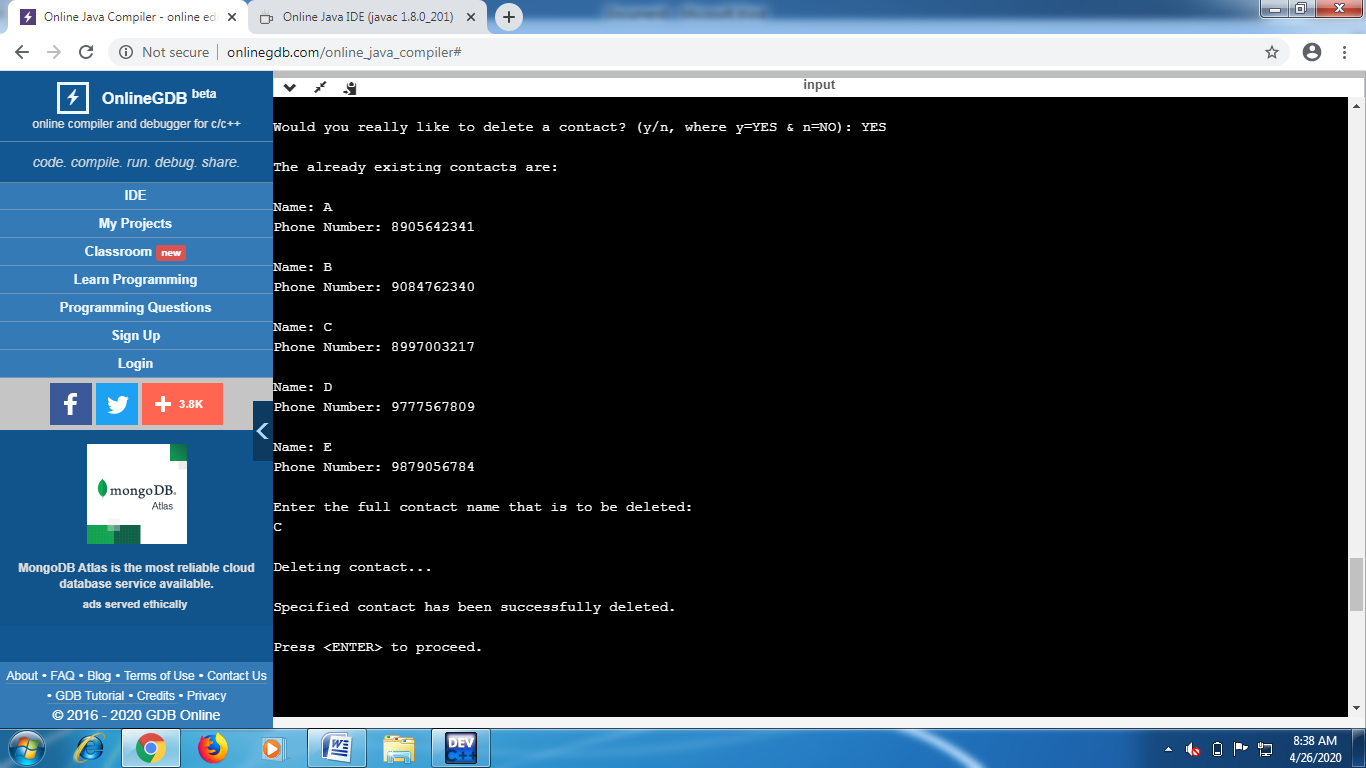
**Fig. 7:** Displaying the Contact List after adding all the contacts



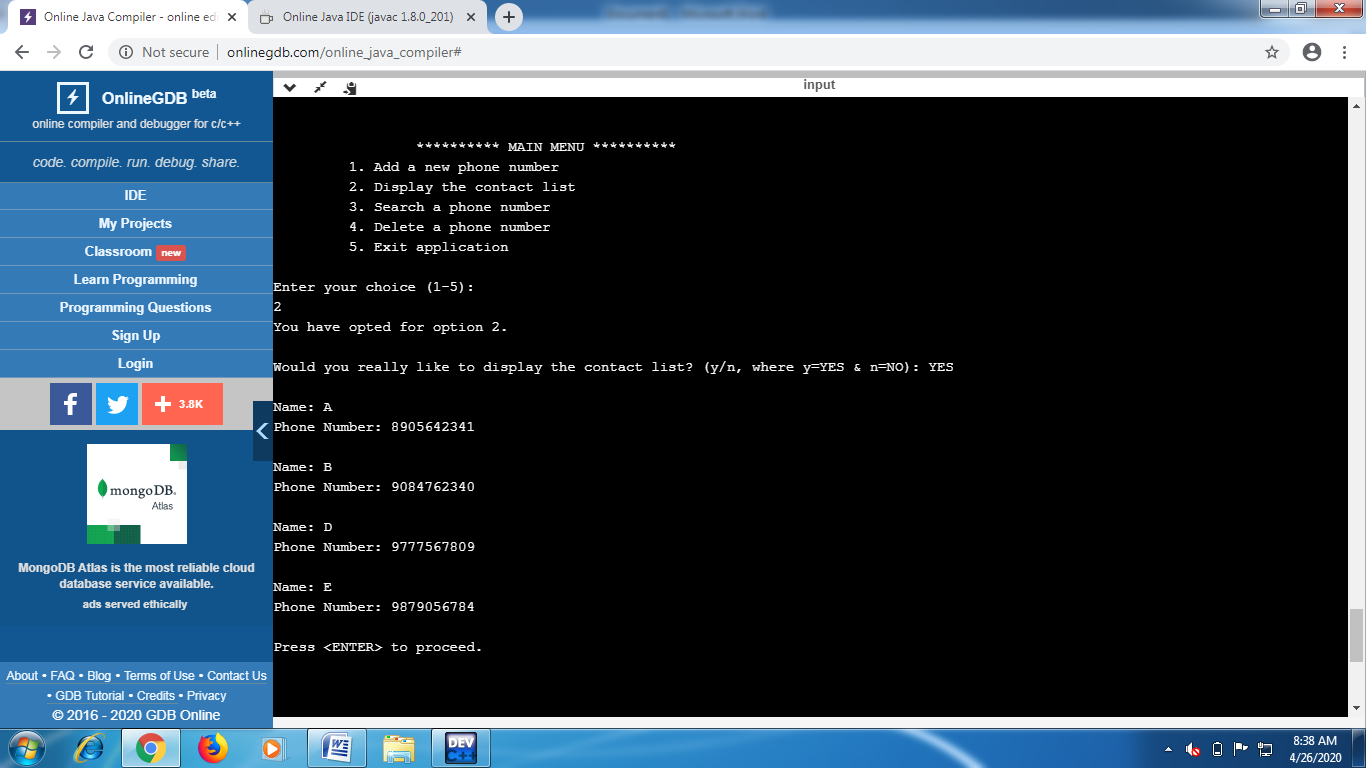
**Fig. 8:** Searching for a contact in the Contact List



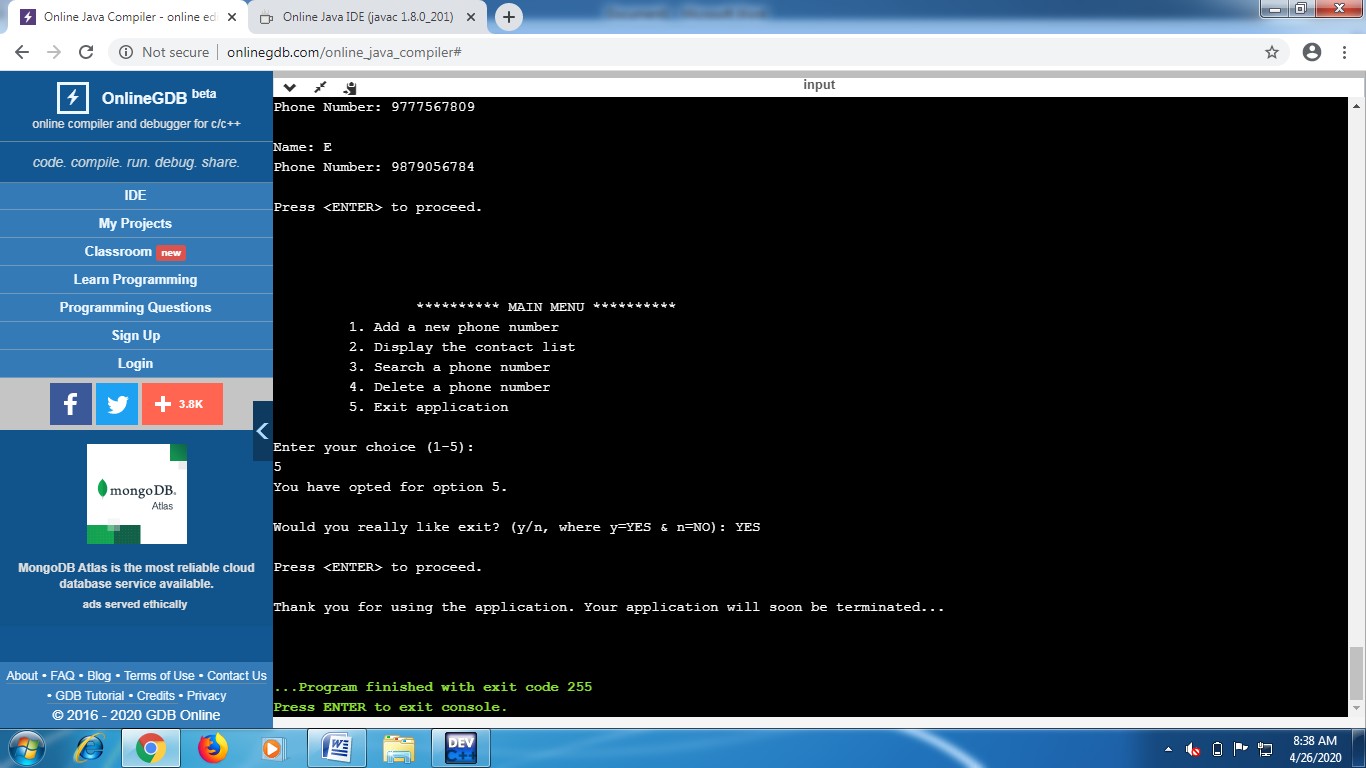
**Fig. 9:** Searching for a contact in the Contact List (contd…)



**Fig. 10:** Deleting a contact from the stored Contacts 



**Fig. 11:** Displaying the Contacts after deleting a contact from the stored Contacts 



**Fig. 12:** Terminating from the Application 