QNO1:

### A)**Base Class and Derived Class with Inheritance:**

using System;

// Base class Telephone

public class Telephone

{

protected string \_phonetype;

// Public method

public void Ring()

{

Console.WriteLine($"Ringing the {\_phonetype}.");

}

}

// Derived class ElectronicPhone

public class ElectronicPhone : Telephone

{

public ElectronicPhone()

{

\_phonetype = "Digital";

}

// Polymorphic method override

public void Ring()

{

Console.WriteLine("Digital phone is ringing.");

}

public void Run()

{

Ring();

}

}

class Program

{

static void Main(string[] args)

{

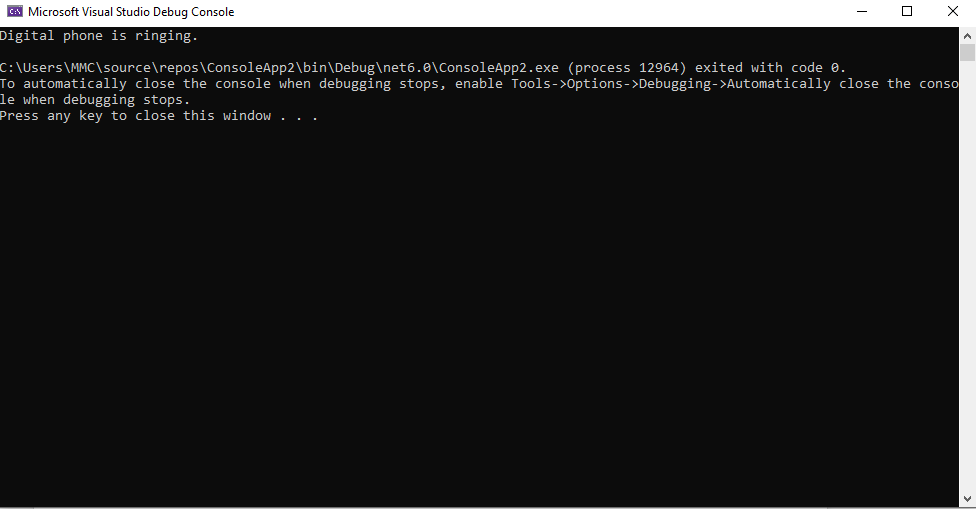
// Create an instance of ElectronicPhone and test the inheritance

ElectronicPhone electronicPhone = new ElectronicPhone();

electronicPhone.Run();

}

}



### **Abstract Class and Derived Classes with Abstract Method:**

using System;

// Abstract base class Telephone

public abstract class Telephone

{

protected string \_phonetype;

// Abstract method

public abstract void Ring();

}

// Derived class DigitalPhone

public class DigitalPhone : Telephone

{

public DigitalPhone()

{

\_phonetype = "Digital";

}

// Override abstract method

public override void Ring()

{

Console.WriteLine("Digital phone is ringing.");

}

}

// Derived class TalkingPhone

public class TalkingPhone : Telephone

{

public TalkingPhone()

{

\_phonetype = "Talking";

}

// Override abstract method

public override void Ring()

{

Console.WriteLine("Talking phone is ringing.");

}

}

class Program

{

static void Main(string[] args)

{

// Create instances of derived classes and test the abstract method

Telephone digitalPhone = new DigitalPhone();

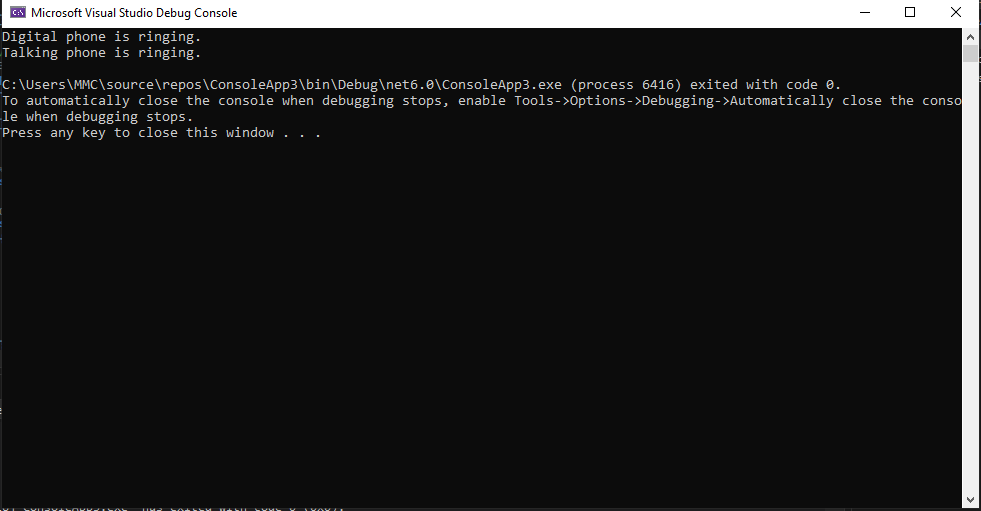
digitalPhone.Ring();

Telephone talkingPhone = new TalkingPhone();

talkingPhone.Ring();

}

}



C)using System;

// Abstract base class Telephone

public abstract class Telephone

{

protected string \_phonetype;

// Abstract method

public abstract void Ring();

}

// Derived class DigitalPhone

public class DigitalPhone : Telephone

{

public DigitalPhone()

{

\_phonetype = "Digital";

}

// Override abstract method

public override void Ring()

{

Console.WriteLine("Digital phone is ringing.");

}

}

// Derived class TalkingPhone

public class TalkingPhone : Telephone

{

public TalkingPhone()

{

\_phonetype = "Talking";

}

// Override abstract method

public override void Ring()

{

Console.WriteLine("Talking phone is ringing.");

}

}

class Program

{

static void Main(string[] args)

{

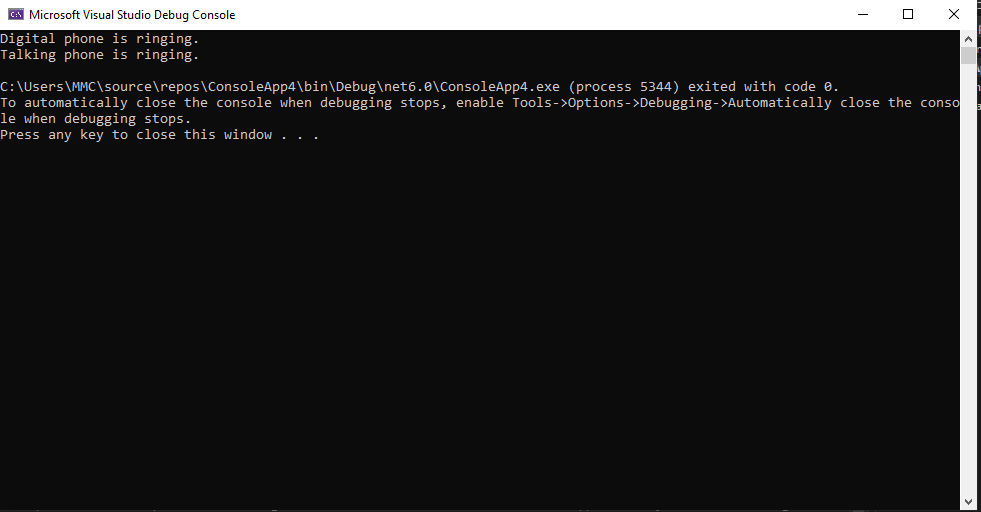
// Create instances of derived classes and test the abstract method

Telephone digitalPhone = new DigitalPhone();

digitalPhone.Ring();

Telephone talkingPhone = new TalkingPhone();

talkingPhone.Ring();



QNO2:

using System;

using System.Collections;

using System.Windows.Forms;

namespace WindowsFormsApp

{

public partial class MainForm : Form

{

private ArrayList itemList = new ArrayList();

public MainForm()

{

InitializeComponent();

}

private void addButton\_Click(object sender, EventArgs e)

{

// Add item to the list

string newItem = itemTextBox.Text;

itemList.Add(newItem);

UpdateListBox();

}

private void removeButton\_Click(object sender, EventArgs e)

{

// Remove selected item from the list

if (itemListBox.SelectedIndex != -1)

{

itemList.RemoveAt(itemListBox.SelectedIndex);

UpdateListBox();

}

}

private void clearButton\_Click(object sender, EventArgs e)

{

// Clear all items from the list

itemList.Clear();

UpdateListBox();

}

private void UpdateListBox()

{

itemListBox.Items.Clear();

foreach (string item in itemList)

{

itemListBox.Items.Add(item);

}

}

}

}