HANDS-ON EXERCISE FOR MID 4

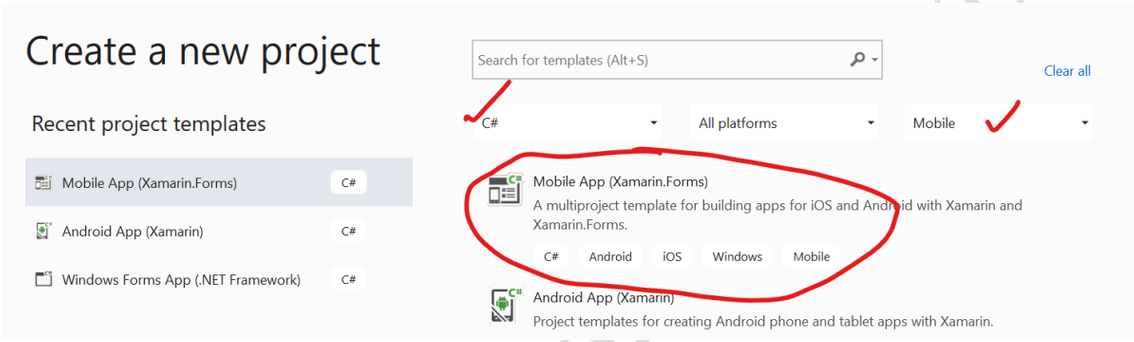
OBJECTIVES:

- Learn how to read from JSON file
- Be able convert JSON data into a ListView

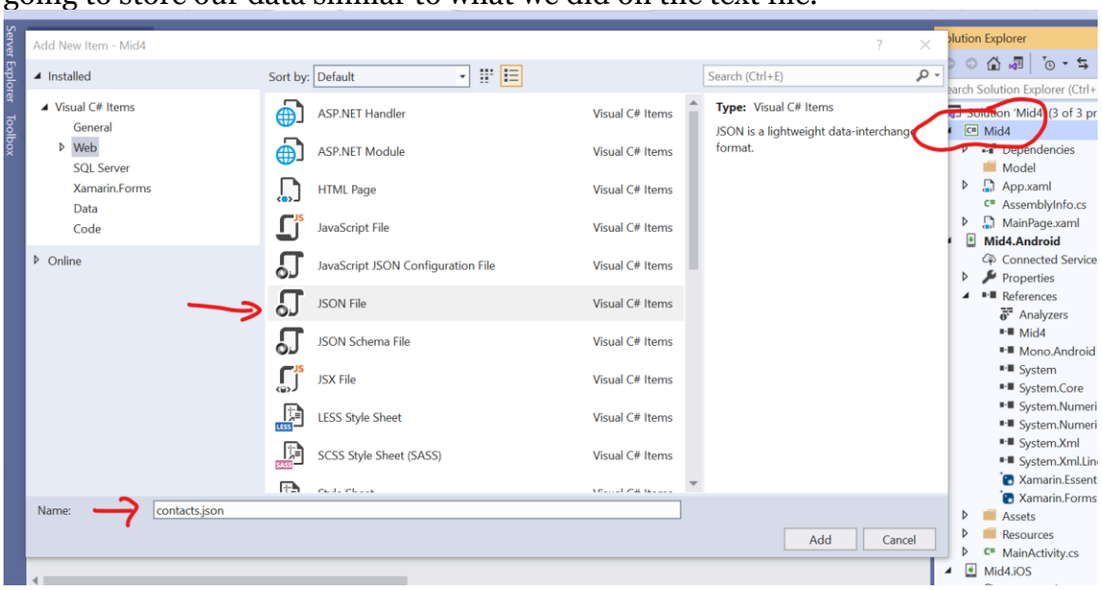
DIRECTIONS:

Create a mobile app that can read data from a JSON file and display it in ListView format.

1. Create a new project in Visual Studio 2019 and select Mobile App (Xamarin.Forms) on the template options. **Save it as Mid4_YOURFULLNAME**.



2. Right click on your main project directory and add a new file. Select Web from the menu on the left, then click on JSON file. Save the file as **contacts.json**. This file is going to store our data similar to what we did on the text file.



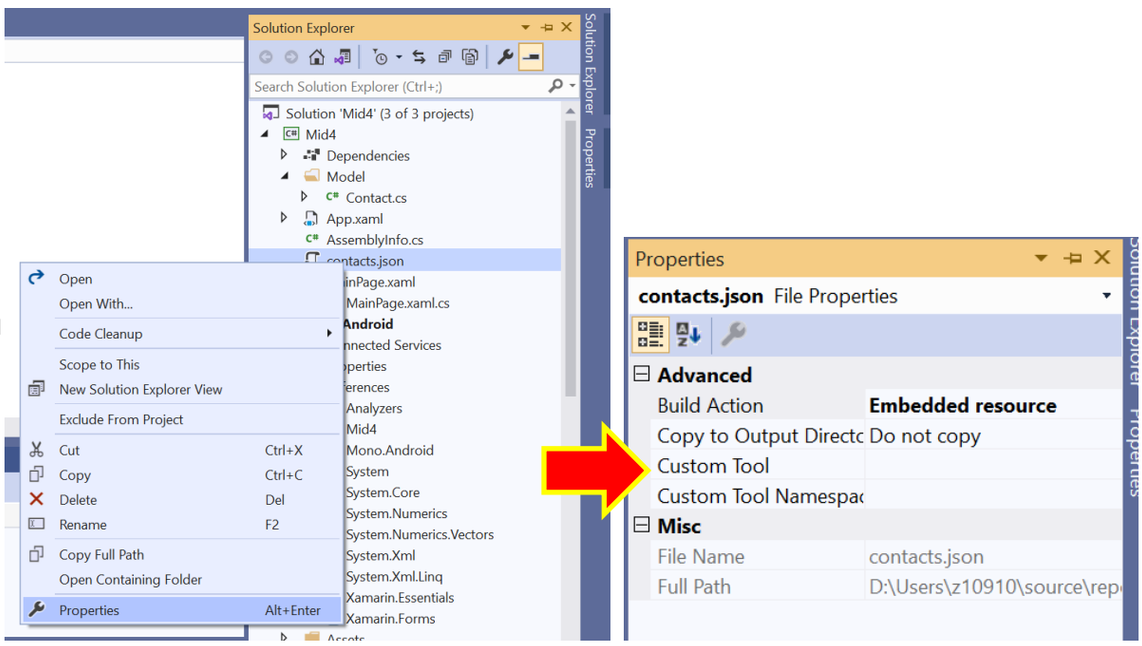
Delete any contents of your contacts.json file, then copy the given codes. JSON files are way better that any text files because it has a much-structured way of storing data.

```
{
  "contacts": [
    {
      "id": "c200",
      "name": "Ravi Tamada",
      "email": "ravi@gmail.com",
      "address": "xx-xx-xxxx,x - street, x - country",
      "gender": "male",
```

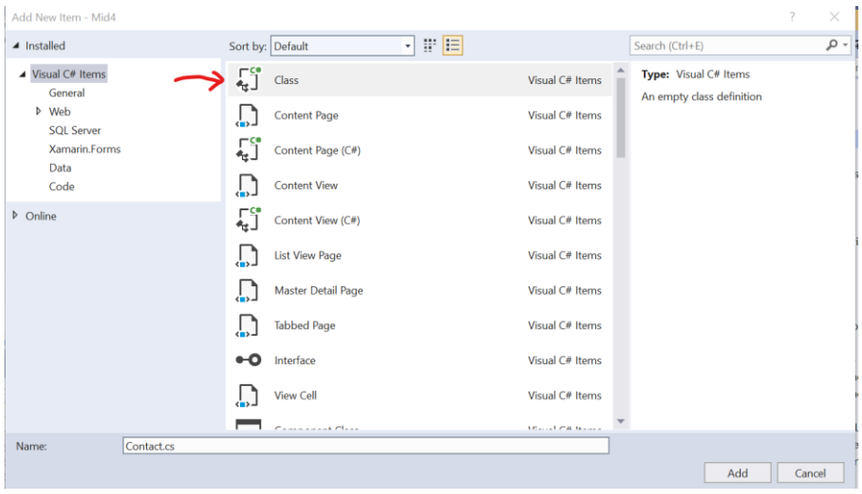
```
      "phone": {
        "mobile": "+91 0000000000",
        "home": "00 000000",
        "office": "00 000000"
      }
    },
    {
      "id": "c201",
      "name": "Johnny Depp",
      "email": "johnny_depp@gmail.com",
      "address": "xx-xx-xxxx,x - street, x - country",
      "gender": "male",
      "phone": {
        "mobile": "+91 0000000000",
        "home": "00 000000",
        "office": "00 000000"
      }
    },
    {
      "id": "c202",
      "name": "Leonardo Dicaprio",
      "email": "leonardo_dicaprio@gmail.com",
      "address": "xx-xx-xxxx,x - street, x - country",
      "gender": "male",
      "phone": {
        "mobile": "+91 0000000000",
        "home": "00 000000",
        "office": "00 000000"
      }

    }
  ]
}
```

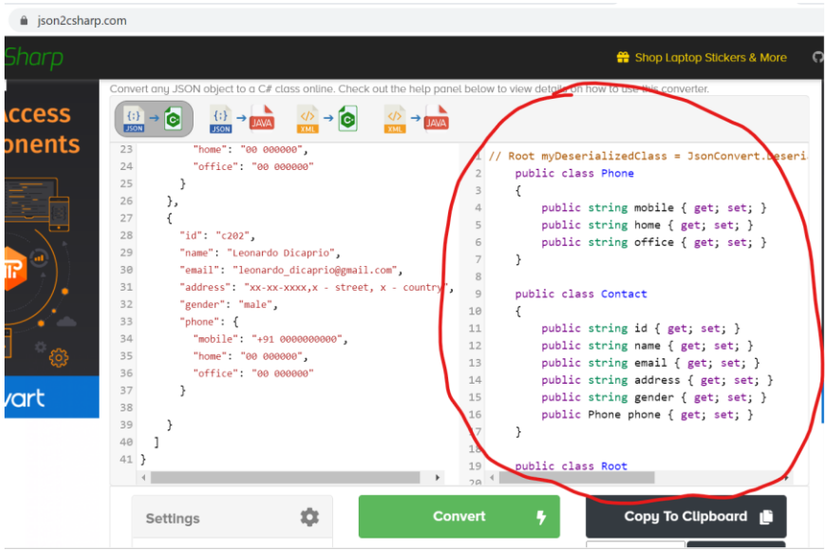
Right click contacts.json on the Solution Explorer, and select Properties. Then on the Build Action option, select "Embedded resource". Embedded Resource means that this file is embedded in the main project build output as a DLL or executable. It is typically used for resource files.



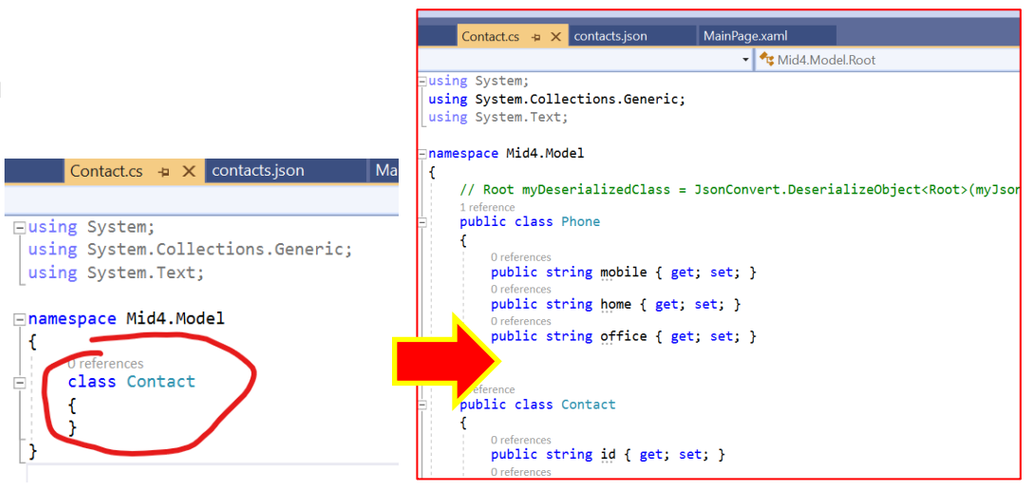
3. In order to create a class that can handle the data properties of the json file, you need to create a folder in your main project and name it as "**Model**". Then right click this folder, add a new item, select Class, and save the file as **Contact.cs**.



Copy the entire contents of your json file, then go to https://json2csharp.com/. Paste the code and press the Convert button. Copy the newly converted code that appears on the website and save it as the content of your **Contacts.cs** file. What we did is to simply convert the JSON file into its equivalent class file which will be used to pickup and reflect data back to your mobile app.



Your class file should now look like the photo on the right side as shown below.



Ensure that the last entry of your Contact.cs class be converted from **Root** to **ContactList**

For clarity purposes, your Contact.cs should contain the following sub-classes and properties.

```
    public class Phone
    {
        public string mobile { get; set; }
        public string home { get; set; }
        public string office { get; set; }
    }

    public class Contact
    {
        public string id { get; set; }
        public string name { get; set; }
        public string email { get; set; }
        public string address { get; set; }
        public string gender { get; set; }
        public Phone phone { get; set; }
    }

    public class ContactList
    {
        public List<Contact> contacts { get; set; }
    }
```

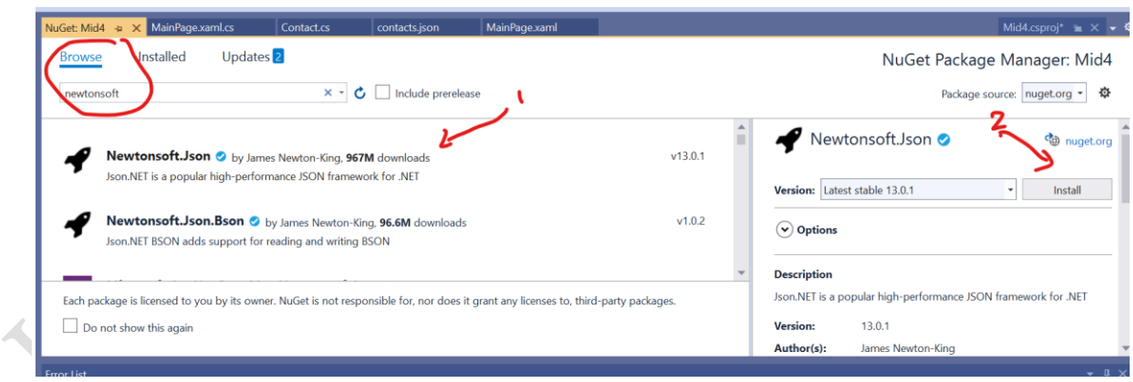
4. Go to your **MainPage.xaml**, remove any code inside the content page and insert the codes below.


```
<Grid>
        <Grid>
            <Grid.RowDefinitions>
                <RowDefinition Height="Auto"/>
                <RowDefinition Height="*"/>
            </Grid.RowDefinitions>
            <Frame BackgroundColor="DarkSeaGreen" Padding="24" CornerRadius="0">
                <Label Text="Contact Us" HorizontalTextAlignment="Center"
TextColor="White" FontSize="28"/>
            </Frame>
```

```
            <ListView x:Name="listviewConacts" Grid.Row="1"
HorizontalOptions="FillAndExpand" Footer="" HasUnevenRows="True">
                <ListView.ItemTemplate>
                    <DataTemplate>
                        <ViewCell>
                            <Grid HorizontalOptions="FillAndExpand" Padding="10">
                                <Grid.RowDefinitions>
                                    <RowDefinition Height="Auto"/>
                                    <RowDefinition Height="Auto"/>
                                    <RowDefinition Height="Auto"/>
                                    <RowDefinition Height="Auto"/>
                                </Grid.RowDefinitions>
                                <Label Text="{Binding name}"
HorizontalOptions="StartAndExpand" Grid.Row="0" TextColor="Blue"
FontAttributes="Bold"/>
                                <Label Text="{Binding email}"
HorizontalOptions="StartAndExpand" Grid.Row="1" TextColor="Orange"
FontAttributes="Bold"/>
                                <Label Text="{Binding phone.mobile}"
HorizontalOptions="StartAndExpand" Grid.Row="2" TextColor="Gray"
FontAttributes="Bold"/>
                                <BoxView HeightRequest="2" Margin="0,10,10,0"
BackgroundColor="Gray" Grid.Row="3" HorizontalOptions="FillAndExpand" />
                            </Grid>
                        </ViewCell>

                    </DataTemplate>
                </ListView.ItemTemplate>
            </ListView>
        </Grid>
    </Grid>
```

5. Right click on your main project folder and select "**Manage Nuget Packages**". This package is useful in managing the content of any JSON file. On the Browse tab, type newtonsoft and press the enter key. Then click on the latest version that appears and press the Install button. Then press OK on the Preview Changes dialog to initiate the installation.



6. Go to your **MainPage.xaml.cs** and insert the given method below the public Mainpage() constructor. This method reads the content of the JSON file, converts it into a list, then display results into a ListView.

```
void GetJsonData()
        {
            string jsonFileName = "contacts.json";
            ContactList ObjContactList = new ContactList();
            var assembly = typeof(MainPage).GetTypeInfo().Assembly;
            Stream stream =
assembly.GetManifestResourceStream($"{assembly.GetName().Name}.{jsonFileName}");
            using (var reader = new System.IO.StreamReader(stream))
            {
                var jsonString = reader.ReadToEnd();
```

```
                //Converting JSON Array Objects into generic list
                ObjContactList =
JsonConvert.DeserializeObject<ContactList>(jsonString);
            }
            //Binding listview with json string
            listviewConacts.ItemsSource = ObjContactList.contacts;
        }
```

Also, you need to add the following libraries as highlighted. Below the InitializeComponent(), add the method call **GetJsonData();** as shown in Line 20.

```
7   using Xamarin.Forms;
8   using Mid4.Model; //add reference to your model folder
9   using System.Reflection; //for GetTypeInfo
10  using System.IO; //for Stream reader
11  using Newtonsoft.Json; //for JSON files
12
13  namespace Mid4
14  {
        5 references
15      public partial class MainPage : ContentPage
16      {
            1 reference
17          public MainPage()
18          {
19              InitializeComponent();
20              GetJsonData();
21          }
22
            1 reference
23          void GetJsonData()
24          {
25              string jsonFileName = "contacts.json";
26              ContactList ObjContactList = new ContactList();
27              var assembly = typeof(MainPage).GetTypeInfo().Assembly;
28              Stream stream = assembly.GetManifestResourceStream($"{assembly.GetName().Name}.{jsonFileName}");
29              using (var reader = new System.IO.StreamReader(stream))
30              {
31                  var jsonString = reader.ReadToEnd();
32
33                  //Converting JSON Array Objects into generic list
```

7. Run your application, and output should be similar below.