INFORMATION ENCODING STANDARDS -Assignment:4

1. **Create a Word document programmatically. Start here**
    1. **Use your API data source and output the details of each entry programmatically to the document.**
        1. **For example, output the text "Hello, my name is <name>" to the document area.**

Code used for implementation:

```
WordprocessingDocument wordprocessingDocument = WordprocessingDocument.Open("C:\\Users\\Owner\\Desktop\\man1.docx", true);

// Assign a reference to the existing document body.
MainDocumentPart mainPart = wordprocessingDocument.MainDocumentPart;
Body body = wordprocessingDocument.MainDocumentPart.Document.Body;
// Add new text.
Paragraph para = body.AppendChild(new Paragraph());
Run run = para.AppendChild(new Run());
run.AppendChild(new Text("Hello my name is " + student.FirstName + "   " + student.LastName));
run.AppendChild(new Break() { Type = BreakValues.Page });
wordprocessingDocument.Close();
```

Explanation:

The basic document structure of a WordProcessingML document consists of the **document** and **body** elements, followed by one or more block level elements such as **p**, which represents a paragraph. A paragraph contains one or more **r** elements. The **r** stands for run, which is a region of text with a common set of properties, such as formatting. A run contains one or more **t** elements. The **t** element contains a range of text. The following code example shows the WordprocessingML markup for a document that contains the text "Example text."

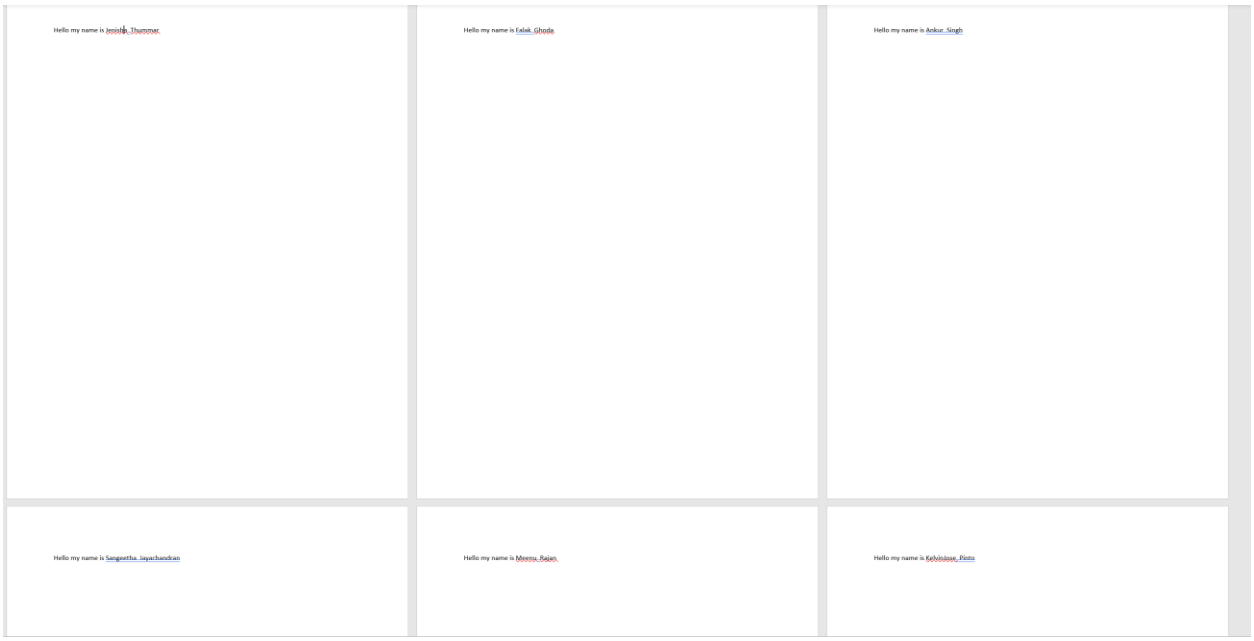
XML Copy

```
<w:document xmlns:w="https://schemas.openxmlformats.org/wordprocessingml/2006/main">
  <w:body>
    <w:p>
      <w:r>
        <w:t>Example text.</w:t>
      </w:r>
    </w:p>
  </w:body>
</w:document>
```

I have recreated the above structure using openxml. An additional page break is added to the run section.

Output:

2.Each entry should have one page



After the execution of the code, we get “Hello my name is <Firstname Lastname>” of the students in each page. Each student has their own page.

3.Add an image to your document area. Use your myimage.jpg from the FTP if your list does not have links to images.

Code used for the implementation:

```
WordprocessingDocument wordprocessingDocument = WordprocessingDocument.Open("C:\\Users\\Owner\\Desktop\\info(1).docx", true);

// Assign a reference to the existing document body.
MainDocumentPart mainPart = wordprocessingDocument.MainDocumentPart;
Body body = wordprocessingDocument.MainDocumentPart.Document.Body;
// Add new text.
Paragraph para = body.AppendChild(new Paragraph());
Run run = para.AppendChild(new Run());
run.AppendChild(new Text("Hello my name is " + student.FirstName + "  " + student.LastName));
run.AppendChild(new Break() { Type = BreakValues.Page });
Student.Graph(wordprocessingDocument, fileName);
wordprocessingDocument.Close();
```

```
1 reference
public static string Graph(WordprocessingDocument wpd, string filepath)
{
    ImagePart ip = wpd.MainDocumentPart.AddImagePart(ImagePartType.Jpeg);
    using (FileStream fs = new FileStream(filepath, FileMode.Open))
    {
        if (fs.Length == 0) return string.Empty;
        ip.FeedData(fs);

    }

    return wpd.MainDocumentPart.GetIdOfPart(ip);

}
```

```
0 references
private static void AddImageToBody(WordprocessingDocument wordDoc, string relationshipId)
{
    // Define the reference of the image.
    var element =
         new Drawing(
             new DW.Inline(
                 new DW.Extent() { Cx = 990000L, Cy = 792000L },
                 new DW.EffectExtent()
                 {
                     LeftEdge = 0L,
                     TopEdge = 0L,
                     RightEdge = 0L,
                     BottomEdge = 0L
                 },
                 new DW.DocProperties()
                 {
                     Id = (UInt32Value)1U,
                     Name = "Picture 1"
                 },
                 new DW.NonVisualGraphicFrameDrawingProperties(
                     new A.GraphicFrameLocks() { NoChangeAspect = true }),
                 new A.Graphic(
                     new A.GraphicData(
                         new PIC.Picture(
                             new PIC.NonVisualPictureProperties(
                                 new PIC.NonVisualDrawingProperties()
                                 {
                                     Id = (UInt32Value)0U,
                                     Name = "abc.jpg"
                                 },
                                 new PIC.NonVisualPictureDrawingProperties()),
                             new PIC.BlipFill(
                                 new A.Blip(
                                     new A.BlipExtensionList(
                                         new A.BlipExtension()
                                         {
                                             Uri =
                                                "{28A0092B-C50C-407E-A947-70E7404B1C1C}"
                                         })
                                 )
                                 {
                                     Embed = relationshipId,
                                     CompressionState =
                                     A.BlipCompressionValues.Print
                                 },
                                 new A.Stretch(
                                     new A.FillRectangle())),
                             new PIC.ShapeProperties(
                                 new A.Transform2D(
                                     new A.Offset() { X = 0L, Y = 0L },
                                     new A.Extents() { Cx = 990000L, Cy = 792000L }),
                                 new A.PresetGeometry(
                                     new A.AdjustValueList()
                                 )
                                 { Preset = A.ShapeTypeValues.Rectangle }))
                         )
                         { Uri = "https://schemas.openxmlformats.org/drawingml/2006/picture" })
                 )
                 {
                     DistanceFromTop = (UInt32Value)0U,
                     DistanceFromBottom = (UInt32Value)0U,
                     DistanceFromLeft = (UInt32Value)0U,
                     DistanceFromRight = (UInt32Value)0U,
                     EditId = "50D07946"
                 });

    // Append the reference to body, the element should be in a Run.
    wordDoc.MainDocumentPart.Document.Body.AppendChild(new Paragraph(new Run(element)));
}
```

4. Save the file programmatically to your FTP directory as *info.docx*.

```
//The below code uploads the csv to my folder.commented to prevent frequent uploads.

string localUploadFilePath = @"C:\Users\Owner\Desktop\info.docx";
string remoteUploadFileDestination = "/200467080 Manoj Gottumukkala/info.docx";
Console.WriteLine(FTP.UploadFile(localUploadFilePath, Constants.FTP.BaseUrl + remoteUploadFileDestination));
```

The above code uploads the file into FTP folder.

2.Create an Excel file programmatically. Start here

1. **Add "Hello, my name is <your name>" to cell A2 of the first sheet.**

Code:

After finding my name using .find, I have used the below code

```
TestModel tm = new TestModel();
tm.intro = "Hello my name is " + student.FirstName + student.LastName;
tmList.testData.Add(tm);
```

1	Introduction
2	Hello my name is Manoj Gottumukkala

Creation of the header:

```
private Row CreateHeaderRowForExcel()
{
    Row workRow = new Row();
    workRow.Append(CreateCell("Introduction"));
    //workRow.Append(CreateCell("Student Id", 2U));
    //workRow.Append(CreateCell("Student Code", 2U));
    //workRow.Append(CreateCell("Student FirstName", 2U));
    //workRow.Append(CreateCell("Student Lastname", 2U));
    //workRow.Append(CreateCell("Student Record", 2U));
    //workRow.Append(CreateCell("Student DateofBirth", 2U));
    //workRow.Append(CreateCell("Student Age", 2U));
    return workRow;
}
```

The detailed code is available in the project.

2.

1. **Use the student list from our class and output to a new sheet using the following column names:**
    1. **StudentId**
    2. **StudentCode**
    3. **FirstName**
    4. **LastName**
    5. **DateOfBirth**
    6. **IsMe (indicates your record with a 1 or 0)**
    7. **Age (calculated formula for age)**
        1. **Refer to this page for information**
        2. **Do not use the C# implementation**

Code for Creation of the header fields:

```
private Row CreateHeaderRowForExcel()
{
    Row workRow = new Row();
    //workRow.Append(CreateCell("Introduction"));
    workRow.Append(CreateCell("Student Id", 2U));
    workRow.Append(CreateCell("Student Code", 2U));
    workRow.Append(CreateCell("Student FirstName", 2U));
    workRow.Append(CreateCell("Student Lastname", 2U));
    workRow.Append(CreateCell("Student Record", 2U));
    workRow.Append(CreateCell("Student DateofBirth", 2U));
    workRow.Append(CreateCell("Student Age", 2U));
    return workRow;
}

private Row GenerateRowForChildPartDetail(TestModel testmodel)
{
    Row tRow = new Row();
    //tRow.Append((CreateCell(testmodel.intro)));
    tRow.Append(CreateCell(testmodel.TestId.ToString()));
    tRow.Append(CreateCell(testmodel.TestCode));
    tRow.Append(CreateCell(testmodel.TestFname));
    tRow.Append(CreateCell(testmodel.TestLname));
    tRow.Append(CreateCell(testmodel.TestRecord.ToString()));
    tRow.Append(CreateCell(testmodel.Testdate));
    tRow.Append(CreateCell(testmodel.age.ToString()));
    return tRow;
}
```

Code for creation of content in the cells:

```
TestModel tm = new TestModel();
tm.TestId = count;
tm.TestCode = student.StudentCode;
tm.TestFname = student.FirstName;
tm.TestLname = student.LastName;

tm.TestRecord = student.Record;
tm.Testdate = student.DateOfBirthString;
try
{
    DateTime oDate = Convert.ToDateTime(tm.Testdate);
    DateTime thisday = DateTime.Today;
    tm.age = (thisday.Year - oDate.Year);
}
catch
{

}


//Adds student to students list.
tmList.testData.Add(tm);
```

Output:

Student Id	Student Code	Student FirstName	Student Lastname	Student Record	Student DateofBirth	Student Age
1	83100937	Leah	Stepanek	0	04/11/1965	55
2	200289345	Carolyn	Knight	0	05/20/1987	33
3	200335615	Chris	Dyck	0	11/29/1971	49
4	200443749	Binaya	Gurung	0	08/06/1995	25
5	200449414	Jiyoung	Sohn	0	08/04/1980	40
6	200449869	Sucheta	Karande	0	04/11/1996	24
7	200450776	Jenisha	Thummar	0	21-10-1998	0
8	200452357	Falak	Ghoda	0	17-04-1998	0
9	200454993	Ankur	Singh	0	01/08/1992	28
10	200454997	Sangeetha	Jayachandran	0	06/25/1998	22
11	200455930	Meenu	Rajan	0		0
12	200458890	KelvinJose	Pinto	0	06/03/1997	23
13	200459092	Rajdeep	Kaur	0	03/03/1998	22
14	200460055	MehmetOzan	Kaya	0	29/06/1995	0
15	200463142	Piyush	Tyagi	0	11/14/1996	24
16	200464602	Dixit	Ohri	0	02/04/1995	25
17	200464960	Shivansh	Desai	0	03/10/1994	26
18	200467080	Manoj	Gottumukkala	1	18/10/97	0
19	200467506	Janina	Cortez	0	11/21/1994	26
20	200467632	Meet	Patel	0	11/12/1998	22
21	200468325	Abdullah	Mohammed	0		2019
22	200468841	AndrewKurian	Jacob	0	05/05/1997	23
23	200469298	Tomasz	Szumski	0	01/07/57	63
24	200470130	Osahon	Ighodaro	0	11 Dec 1986	34
25	200470693	NnennaGrace	Alphonsus	0	12/12/1990	30
26	200471222	Sophie	Kellman	0	19810101	0
27	200471292	Danhong	Tang	0	2000/11/11	20
28	200471940	Karish	Thangarajah	0	3/6/1997	23
29	200473131	Sol	Jun	0	06/30/1997	23
30	200473709	David	Mascioli	0	09/21/1972	48

We can clearly see that record with my name (Manoj Gottumukkala) is set to 1. Age is calculated for the dateofbirth's which are in correct format.

3.Save the file programmatically to your FTP directory as *info.xlsx*.

```
//Uploading the file to FTP folder.
string localUploadFilePath = @"C:\Users\Owner\Desktop\Output1.xlsx";
string remoteUploadFileDestination = "/200467080 Manoj Gottumukkala/info.xlsx";
Console.WriteLine(Model.Utilities.FTP.UploadFile(localUploadFilePath, Constants.FTP.BaseUrl + remoteUploadFileDestination));
```

This is the code used for uploading the file into the FTP.