eBook Gratuit

APPRENEZ openxml

eBook gratuit non affilié créé à partir des **contributeurs de Stack Overflow.**

#openxml

Table des matières

À propos

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: openxml

It is an unofficial and free openxml ebook created for educational purposes. All the content is extracted from Stack Overflow Documentation, which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official openxml.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapitre 1: Démarrer avec openxml

Remarques

Cette section fournit une vue d'ensemble de ce qu'est openxml et pourquoi un développeur peut vouloir l'utiliser.

Il devrait également mentionner tous les grands sujets dans openxml, et établir un lien avec les sujets connexes. La documentation pour openxml étant nouvelle, vous devrez peut-être créer des versions initiales de ces rubriques connexes.

Examples

Installation d'OpenXML SDK et outil de productivité sur votre ordinateur

Accédez au lien Microsoft pour le téléchargement du SDK OpenXML . Cliquez sur le bouton de téléchargement rouge. Sur l'écran suivant, cliquez sur la case à côté d'OpenXMLSDKToolV25.msi et cliquez sur suivant pour commencer le téléchargement.

Une fois le téléchargement terminé, lancez OpenXMLSDKToolV25.msi et suivez les instructions à l'écran.

Le programme d'installation place les fichiers dans le répertoire par défaut suivant:

```
"C:\Program Files (x86)\Open XML SDK\V2.5"
```

Dans ce répertoire, vous trouverez un fichier Readme qui explique comment utiliser le SDK et un fichier Lisez-moi pour l'outil de productivité.

Créer une nouvelle feuille de calcul avec OpenXML

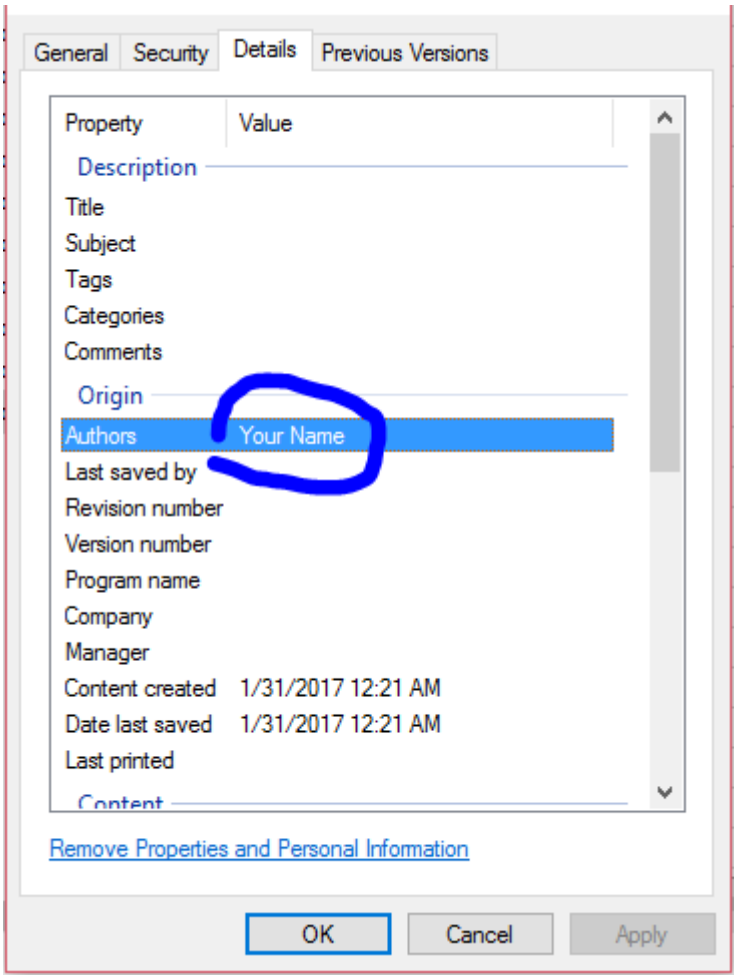
Cette méthode créera une nouvelle feuille de calcul Excel. `fileName` le nom de fichier qui est un nom de chemin de fichier complet.

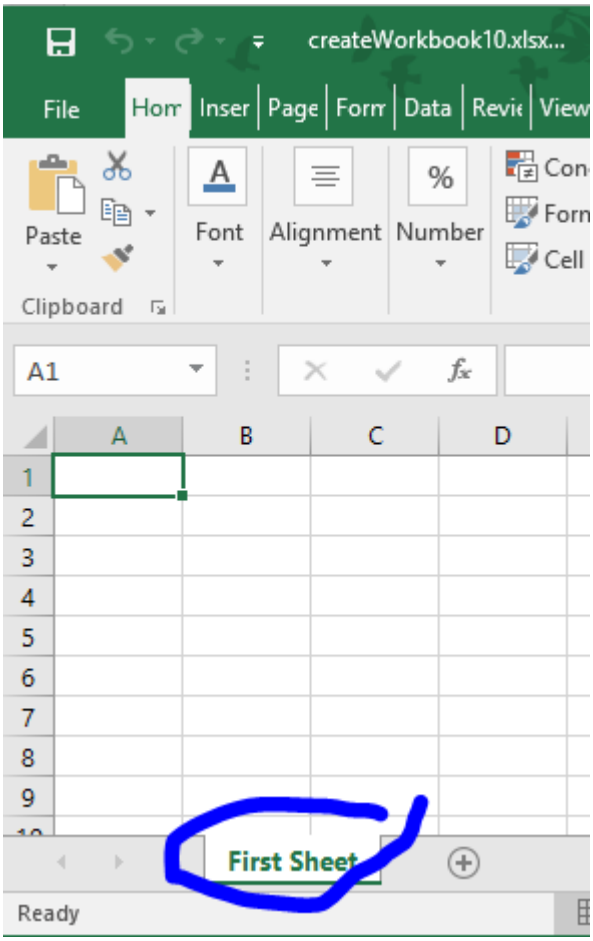
```
using DocumentFormat.OpenXml;
using DocumentFormat.OpenXml.Packaging;
using DocumentFormat.OpenXml.Spreadsheet;
using System;
....
    void Create(string fileName)
    {

        using (SpreadsheetDocument document = SpreadsheetDocument.Create(fileName,
SpreadsheetDocumentType.Workbook))
        {
            var relationshipId = "rId1";

            //build Workbook Part
            var workbookPart = document.AddWorkbookPart();
```

```
            var workbook = new Workbook();
            var sheets = new Sheets();
            var sheet1 = new Sheet() { Name = "First Sheet", SheetId = 1, Id = relationshipId
};
            sheets.Append(sheet1);
            workbook.Append(sheets);
            workbookPart.Workbook = workbook;

            //build Worksheet Part
            var workSheetPart = workbookPart.AddNewPart<WorksheetPart>(relationshipId);
            var workSheet = new Worksheet();
            workSheet.Append(new SheetData());
            workSheetPart.Worksheet = workSheet;

            //add document properties
            document.PackageProperties.Creator = "Your Name";
            document.PackageProperties.Created = DateTime.UtcNow;

        }
```

Pour ce projet, assurez-vous d'inclure la référence à `DocumentFormat.OpenXml` . Cela se trouve dans le chemin spécifié dans l'exemple OpenXML.

La feuille de calcul sera créée avec **votre nom** comme auteur et la première feuille de calcul nommée **première feuille** .



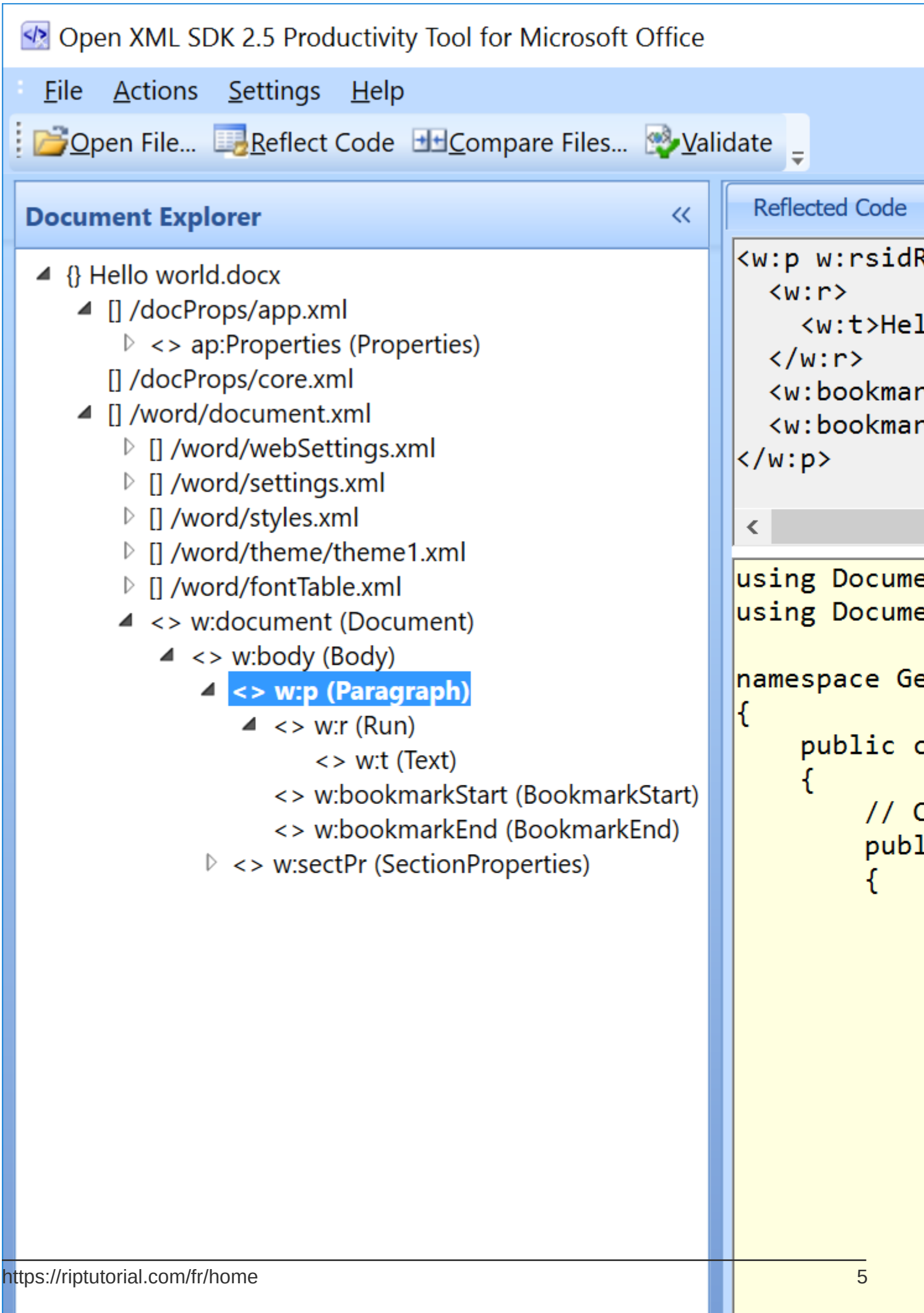


Utilisation de l'outil de productivité Open XML SDK 2.5

La lecture de la spécification pour les formats de document dans OpenXML peut prendre du temps. Parfois, vous voulez simplement voir comment produire une certaine fonctionnalité dans un document Word. L'outil de productivité Open XML SDK 2.5 pour Microsoft Office (OpenXmlSdkTool.exe) est là pour ça. Ses principales caractéristiques sont les suivantes:

- Voir la structure d'un fichier - quelles parties xml contiennent-elles?
- Naviguer dans le xml dans chacune de ces parties
- Générer le code c # pour produire la partie sélectionnée du document
- Lien vers la spécification du format de fichier décrivant plus de détails
- Validation du document OpenXML

Pour un simple 'Hello world.docx', cela ressemble à ceci:


Open XML SDK 2.5 Productivity Tool for Microsoft Office
File
Actions
Settings
Help
Open File...
Reflect Code
Compare Files...
Validate
Document Explorer
{} Hello world.docx
[] /docProps/app.xml
<> ap:Properties (Properties)
[] /docProps/core.xml
[] /word/document.xml
[] /word/webSettings.xml
[] /word/settings.xml
[] /word/styles.xml
[] /word/theme/theme1.xml
[] /word/fontTable.xml
<> w:document (Document)
<> w:body (Body)
<> w:p (Paragraph)
<> w:r (Run)
<> w:t (Text)
<> w:bookmarkStart (BookmarkStart)
<> w:bookmarkEnd (BookmarkEnd)
<> w:sectPr (SectionProperties)
Reflected Code
<w:p w:rsidR
<w:r>
<w:t>Hel
</w:r>
<w:bookmar
<w:bookmar
</w:p>
using Docume
using Docume
namespace Ge
{
public c
{
// C
publ
{

Le volet de gauche montre la structure du document. Le volet supérieur droit affiche le xml correspondant à la sélection dans l'arborescence et, enfin, le volet inférieur droit affiche un code généré pour générer le xml affiché au-dessus.

Cela permet une recherche très pratique sur une fonctionnalité donnée:

- Produire un exemple de document (fx un document Word)
- Ouvrez le document dans l'outil de productivité
- Utilisez 'Reflect Code' pour générer du code

Le SDK peut être téléchargé à l' adresse https://www.microsoft.com/en-us/download/details.aspx?id=30425 - téléchargez et installez les deux packages msi. Après l'installation, utilisez OpenXMLSdkTool.exe installé dans "C: \ Program Files (x86) \ Ouvrir XML SDK \ V2.5 \ tool".

Lire Démarrer avec openxml en ligne: https://riptutorial.com/fr/openxml/topic/6967/demarrer-avec-openxml

Chapitre 2: Comment ajouter une image à un document Word.

Introduction

L'insertion d'une image dans un document Word à l'aide d'OpenXml nécessite deux actions: ajouter l'image dans le fichier openxml et faire référence à l'image dans votre document

Remarques

Si vous ajoutez uniquement l'image à la structure openxml sans la renvoyer dans le document Word, la prochaine fois que vous "ouvrirez / enregistrez" votre document, le fichier image sera supprimé.

Word supprimer toutes les références orphelines. Veillez donc à ajouter l'image dans le document Word, sinon vous devrez recommencer toutes les étapes.

Examples

Ajouter l'image à la structure OpenXml

```
private string AddGraph(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);
}
```

Dans ce cas, nous utilisons un FileStream pour récupérer l'image, mais feedData (Stream) attend n'importe quel type de flux.

Reportez-vous à l'image dans le document Word

```
private void InsertImage(WordprocessingDocument wpd, OpenXmlElement parent, string filepath)
{
    string relationId = AddGraph(wpd, filepath);
    if (!string.IsNullOrEmpty(relationId))
    {
        Size size = new Size(800, 600);

        Int64Value width = size.Width * 9525;
        Int64Value height = size.Height * 9525;
```

```
        var draw = new Drawing(
            new DW.Inline(
                new DW.Extent() { Cx = width, Cy = height },
                new DW.EffectExtent()
                {
                    LeftEdge = 0L,
                    TopEdge = 0L,
                    RightEdge = 0L,
                    BottomEdge = 0L
                },
                new DW.DocProperties()
                {
                    Id = (UInt32Value)1U,
                    Name = "my image name"
                },
                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 = relationId
                                },
                                new PIC.NonVisualPictureDrawingProperties()),
                                new PIC.BlipFill(
                                    new A.Blip(
                                        new A.BlipExtensionList(
                                            new A.BlipExtension() { Uri = "{28A0092B-C50C-
407E-A947-70E740481C1C}" })
                                        )
                                    {
                                        Embed = relationId,
                                        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 = width, Cy = height
}),
                                                    new A.PresetGeometry(new
A.AdjustValueList()) { Preset = A.ShapeTypeValues.Rectangle }))) { Uri =
"http://schemas.openxmlformats.org/drawingml/2006/picture" })
                                                    )
            {
                DistanceFromTop = (UInt32Value)0U,
                DistanceFromBottom = (UInt32Value)0U,
                DistanceFromLeft = (UInt32Value)0U,
                DistanceFromRight = (UInt32Value)0U,
                EditId = "50D07946"
            });

        parent.Append(draw);
    }
}
```

Dans cet exemple, j'ai défini une taille statique de 800 * 600 mais vous pouvez définir la taille dont vous avez besoin

Lire Comment ajouter une image à un document Word. en ligne: https://riptutorial.com/fr/openxml/topic/9397/comment-ajouter-une-image-a-un-document-word-

Chapitre 3: Créer un nouveau document Word avec Open XML

Introduction

La norme de balisage de documents OpenXML est un format basé sur XML qui permet des solutions sur de nombreuses plates-formes logicielles et systèmes d'exploitation.

Examples

Bonjour le monde

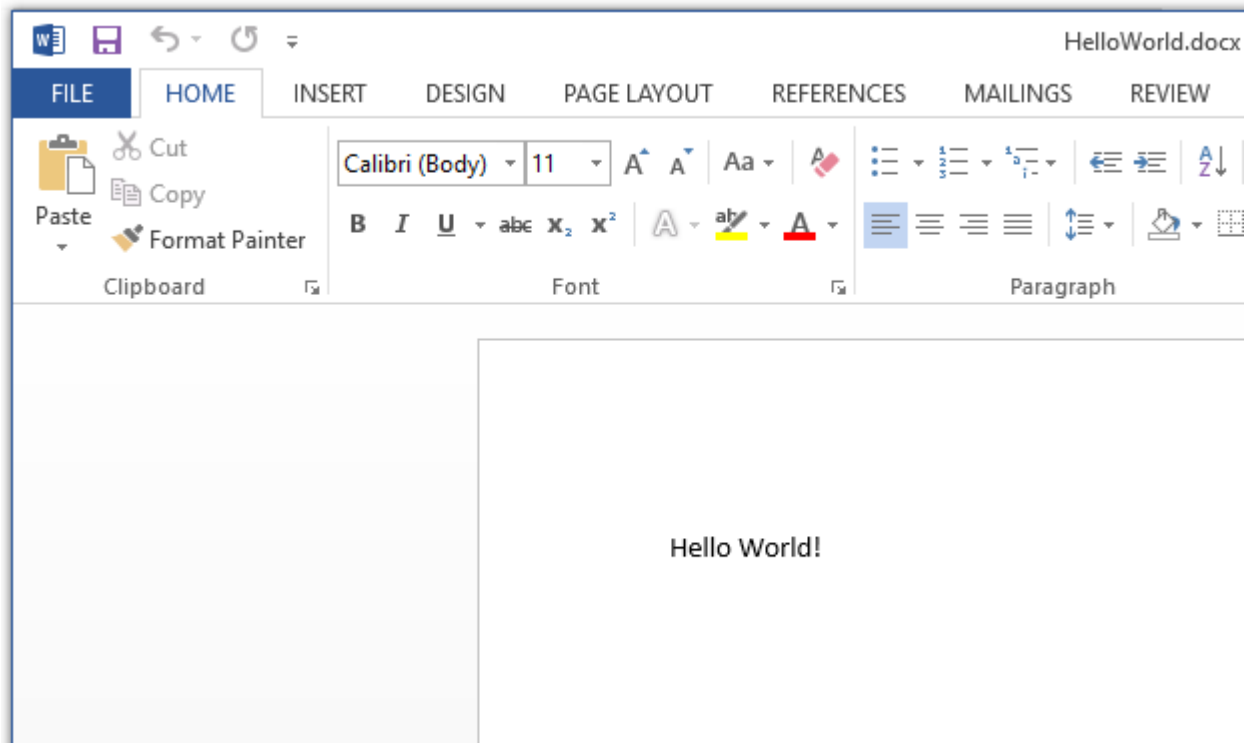
Tout d'abord, créez un nouveau projet de console à l'aide de Visual Studio et ajoutez les fichiers .dll suivants à votre projet:

```
DocumentFormat.OpenXml
WindowsBase
```

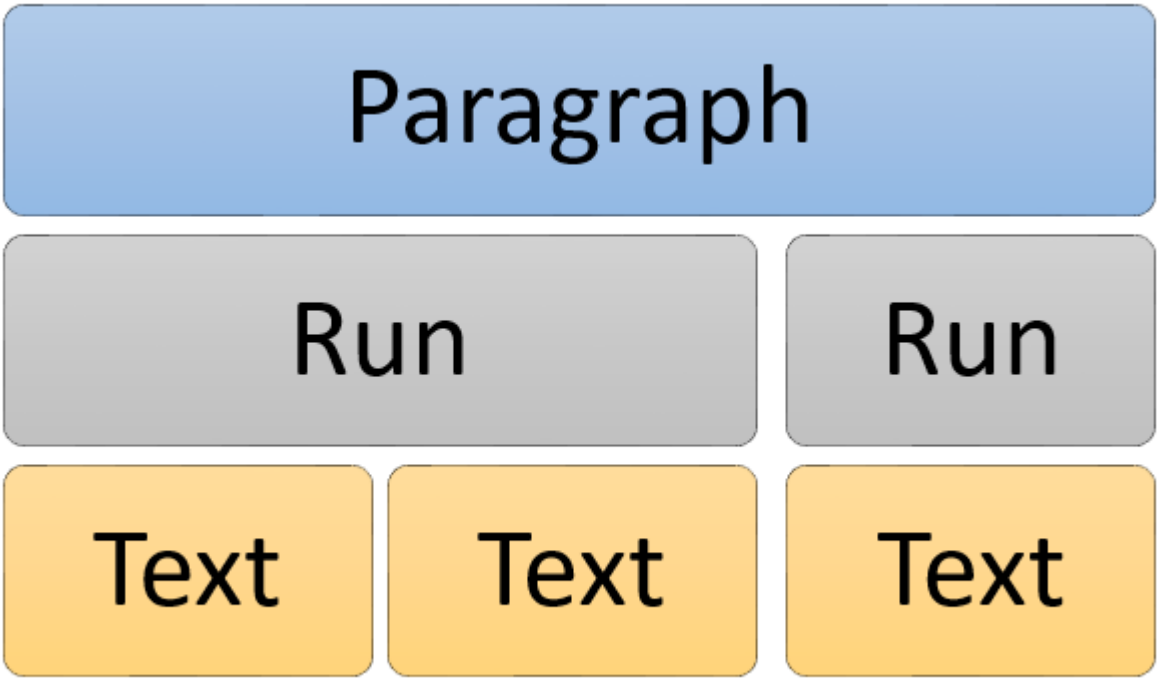
Ensuite, compilez et exécutez le code suivant:

```
static void Main(string[] args)
{
    // Create a Wordprocessing document.
    using ( WordprocessingDocument package = WordprocessingDocument.Create("HelloWorld.docx",
WordprocessingDocumentType.Document))
    {
        // Add a new main document part.
        package.AddMainDocumentPart();

        // Create the Document DOM.
        package.MainDocumentPart.Document =
            new Document(
                new Body(
                    new Paragraph(
                        new Run(
                            new Text("Hello World!")))));

        // Save changes to the main document part.
        package.MainDocumentPart.Document.Save();
    }
}
```

Dans votre dossier `\bin\Debug` , vous devez avoir votre premier document WordprocessingML:



Le texte que nous avons ajouté dans l'exemple ci-dessus est stocké sous la partie principale du document. Dans la partie principale du document se trouve l'élément **document** qui permet à un **corps d'** élément enfant de stocker le texte qui fait notre document. Il existe deux groupes principaux de contenu pour le corps du document, le niveau de bloc *(paragraphes et tableaux)* et le contenu en ligne *(exécutions et texte)* . Le contenu au niveau du bloc fournit la structure principale et contient du contenu en ligne. Pour comprendre l'exemple ci-dessus, nous devons d'abord comprendre la hiérarchie du texte dans WordprocessingML. Un paragraphe est divisé en différentes séries. Un run est l'élément de niveau le plus bas auquel la mise en forme peut être appliquée. Le cycle est à nouveau divisé en différents éléments de texte.



Lire Créer un nouveau document Word avec Open XML en ligne:
https://riptutorial.com/fr/openxml/topic/9833/creer-un-nouveau-document-word-avec-open-xml

Chapitre 4: Insérer une image dans une "forme en ligne" intégrée dans des documents Word

Introduction

Insérez une image dans un document MS Word, par exemple des rectangles et des ovales.

Cette documentation suppose que vous savez comment insérer une image dans un document Word, ouvrir et fermer un document Word à l'aide d'OpenXML.

Examples

Ajoutez les espaces de noms OpenXML suivants à votre classe

```
using System;
using System.Collections.Generic;
using System.Linq;
using DocumentFormat.OpenXml;
using A = DocumentFormat.OpenXml.Drawing;
using DW = DocumentFormat.OpenXml.Drawing.Wordprocessing;
using PIC = DocumentFormat.OpenXml.Drawing.Pictures;
using DocumentFormat.OpenXml.Drawing.Wordprocessing;
using Wps = DocumentFormat.OpenXml.Office2010.Word.DrawingShape;
```

Ouvrez le document et ajoutez l'objet imagePart pour référencer l'image que vous souhaitez insérer dans la forme.

Maintenant, ouvrez le document en utilisant OpenXML, vous devez ajouter une imagePart qui référence l'objet image à l'objet MainDocumentPart en utilisant un flux de fichiers et obtenir l'ID de l'image

```
    string temp;
MainDocumentPart mainPart = document.MainDocumentPart;
                                ImagePart imagePart = mainPart.AddImagePart(ImagePartType.Bmp);

                                using (FileStream stream = new FileStream(barcodepath,
FileMode.Open))
                                {
                                   imagePart.FeedData(stream);
                                }

                                temp = mainPart.GetIdOfPart(imagePart);
```

Obtenir une référence d'un objet Blip

Au bureau OpenXML, une image insérée dans un document Word est considérée comme un objet ou un élément "Blip". La classe est dérivée de DocumentFormat.OpenXml.Drawing the Blip doit avoir une valeur Embed qui est un ID imagePart. L'objet Blip se trouve alors dans un objet / élément BlipFill , qui entre également dans un objet / élément graphicData et qui, à son tour, entre dans un élément d'objet graphique . Je suis à peu près sûr que vous vous êtes rendu compte que tout fonctionne comme un arbre XML. Exemple d'arborescence XML ouverte ci-dessous.

```
<a:graphic xmlns:a="http://schemas.openxmlformats.org/drawingml/2006/main">
                        <a:graphicData
uri="http://schemas.microsoft.com/office/word/2010/wordprocessingShape">
                          <wps:wsp>
                            <wps:cNvSpPr>
                              <a:spLocks noChangeArrowheads="1" />
                            </wps:cNvSpPr>
                            <wps:spPr bwMode="auto">
                              <a:xfrm>
                                <a:off x="0" y="0" />
                                <a:ext cx="1234440" cy="1234440" />
                              </a:xfrm>
                              <a:prstGeom prst="roundRect">
                                <a:avLst>
                                  <a:gd name="adj" fmla="val 16667" />
                                </a:avLst>
                              </a:prstGeom>
                              <a:blipFill dpi="0" rotWithShape="1">
                                <a:blip r:embed="Raade88ffea8d4c1b" />
                                <a:stretch>
                                  <a:fillRect l="10000" t="10000" r="10000" b="10000" />
                                </a:stretch>
                              </a:blipFill>
                            </wps:spPr>
                            <wps:bodyPr rot="0" vert="horz" wrap="square" lIns="91440"
tIns="45720" rIns="91440" bIns="45720" anchor="t" anchorCtr="0" upright="1">
                              <a:noAutofit />
                            </wps:bodyPr>
                          </wps:wsp>
                        </a:graphicData>
                      </a:graphic>
```

Ajout de la référence de l'image aux formes du document modèle.

Maintenant, vous avez une référence de l'image. Insérez l'image dans les formes du document modèle. Pour ce faire, vous devrez utiliser LINQ pour parcourir le document et obtenir une référence à toutes les formes du document. L'élément wps: spPr que vous voyez dans le code XML ci-dessus est l'élément xml des formes d'un document. La classe C # équivalente est WordprocessingShape

```
IEnumerable<DocumentFormat.OpenXml.Office2010.Word.DrawingShape.WordprocessingShape> shapes2 =
document.MainDocumentPart.document.Body.Descendants<DocumentFormat.OpenXml.Office2010.Word.DrawingShape
```

Avec une référence de collection de toutes les formes, parcourez la collection en boucle.

Maintenant que vous avez une collection de toutes les références de forme dans le document. Faites une boucle dans la collection avec un foreach, et à travers chaque itération, créez un objet Blip. Définissez la valeur d'intégration d'objet Blip sur la référence d'ID d'image que vous avez capturée précédemment dans la partie d'image. Créez également un objet Stretch et un objet FillRectangle (ceux-ci ne sont pas vraiment nécessaires, ils ne sont utilisés que pour un alignement correct de l'image). Et ajoutez chacun à son objet parent comme l'équivalent d'arborescence XML.

```
foreach (DocumentFormat.OpenXml.Office2010.Word.DrawingShape.WordprocessingShape sp in
shapes2)
                                {
                                    //Wps.ShapeProperties shapeProperties1 =
                                    A.BlipFill blipFill1 = new A.BlipFill() { Dpi =
(UInt32Value)0U, RotateWithShape = true };
                                    A.Blip blip1 = new A.Blip() { Embed = temp };

                                    A.Stretch stretch1 = new A.Stretch();
                                    A.FillRectangle fillRectangle1 = new A.FillRectangle() {
Left = 10000, Top = 10000, Right = 10000, Bottom = 10000 };
                                    Wps.WordprocessingShape wordprocessingShape1 = new
Wps.WordprocessingShape();



                                        stretch1.Append(fillRectangle1);
                                        blipFill1.Append(blip1);
                                        blipFill1.Append(stretch1);
                                        Wps.ShapeProperties shapeProperties1 =
sp.Descendants<Wps.ShapeProperties>().First();
                                        shapeProperties1.Append(blipFill1);

                                }
```

Lire Insérer une image dans une "forme en ligne" intégrée dans des documents Word en ligne: https://riptutorial.com/fr/openxml/topic/9660/inserer-une-image-dans-une--forme-en-ligne--integree-dans-des-documents-word

Crédits

S. No	Chapitres	Contributeurs
1	Démarrer avec openxml	Community, Daniel Brixen, Mandar Badve, Taterhead
2	Comment ajouter une image à un document Word.	Maxime Porté
3	Créer un nouveau document Word avec Open XML	FortyTwo
4	Insérer une image dans une "forme en ligne" intégrée dans des documents Word	Dankwansere