

UNIVERSITÉ DE SHERBROOKE
DÉPARTEMENT D'INFORMATIQUE

IFT 159

Devoir #2 — Automne 2008

Programmation

Devoir à remettre au plus tard le 25 septembre 2009 à 23h59. Vous pouvez le remettre en classe ou dans le casier identifié «ift159» qui se trouve au sous-sol du D4 en face de la salle D4-0014.

Le but de ce travail pratique est de faire l'analyse d'un problème simple, de s'initier à la décomposition d'un problème, de faire la conception de sa solution à l'aide d'un diagramme structurel et d'en produire un algorithme. **Vous devez utiliser seulement les notions vues dans le cours (fonctions). Vous ne devez pas utiliser la sélection (if).**

Sujet : Document OpenOffice pour analyse de coût.

Un employé du service d'aide à la vie étudiante décide de faire une série d'enquêtes pour connaître les dépenses engagées par certains étudiants¹ pendant une session d'étude à l'université. Il choisit donc, pour chaque enquête, trois étudiants pour lesquels il s'informe de leurs dépenses en terme de logement, d'épicerie, de transport et de loisirs. Les résultats de l'enquête sont reçus dans un fichier non formaté contenant, pour chaque étudiant, son nom et les montants engagés pour chacune des dépenses. Le format de ce fichier est donné à la figure 1.

La personne responsable veut avoir les données dans un format plus lisible. Il désire donc produire un document de format « ODF » (Open Document Format de **OpenOffice**) contenant un tableau pour les résultats des étudiants qui participent à l'enquête. Pour chacune des enquêtes, on veut produire un document « ODF ». Le tableau de classement contiendra, sur chaque ligne, le nom de l'étudiant, les dépenses pour chacune des catégories et le total des dépenses pour chaque étudiant. On veut aussi avoir une dernière ligne contenant la moyenne pour chacune des dépenses ainsi que la moyenne du « total des dépenses des trois étudiants ».

On vous demande donc de développer un programme qui reçoit les résultats de la

¹J'utilise le masculin partout pour ne pas alourdir le texte, mais soyez conscients que le terme « étudiants » désigne aussi les étudiantes.

```

« Matricule de l'étudiant 1 »
« Dépenses en logement de l'étudiant 1 »
« Dépenses en épicerie de l'étudiant 1 »
« Dépenses en loisirs de l'étudiant 1 »
« Dépenses en transports de l'étudiant 1 »

« Matricule de l'étudiant 2 »
« Dépenses en logement de l'étudiant 2 »
« Dépenses en épicerie de l'étudiant 2 »
« Dépenses en loisirs de l'étudiant 2 »
« Dépenses en transports de l'étudiant 2 »

« Matricule de l'étudiant 3 »
« Dépenses en logement de l'étudiant 3 »
« Dépenses en épicerie de l'étudiant 3 »
« Dépenses en loisirs de l'étudiant 3 »
« Dépenses en transports de l'étudiant 3 »

```

FIG. 1 – Format du fichier contenant les résultats des devoirs des étudiants.

compétition en entrée (à partir du fichier) et qui génère automatiquement en sortie un fichier en format « ODF ». Le fichier de sortie doit avoir un format bien précis. Vous trouverez à la figure 2 un exemple d’affichage des résultats dans OpenOffice. Ce fichier de sortie contient :

1. le titre sur trois lignes donnant
 - (a) le nom de l’enquête sur la première ligne,
 - (b) l’année de l’enquête sur la seconde ligne,
 - (c) le numéro de l’enquête sur la troisième ligne;
2. le tableau qui contient :
 - une ligne titre,
 - une ligne pour chaque étudiant,
 - une ligne contenant des statistiques.

L’entête sera composé d’un texte centré en gras de 18 points. Le tableau sera centré, aura une largeur de 15cm et aura six colonnes de largeur égale de 2.5 cm chacune.

L’usager du programme doit spécifier en entrée le nom de l’enquête, le numéro de l’enquête, l’année, le numéro et le nom du fichier contenant les résultats de l’enquête. Le nom du fichier de sortie, en format « ODF », devra obligatoirement porter le nom « **content.xml** ». Le programme devra lire le contenu du fichier, produire un second fichier contenant le document « ODF » (extension « .odt ») et afficher le nombre de lignes générées dans le document « ODF ».

**Enquêtes sur les dépenses étudiantes
2009
Résultats pour enquête #4.**

Nom de l'étudiant	Logement	Épicerie	Loisirs	Transport	Totaux
Fred cailloux	900	2000	1000	300	4200
Arthur Laroche	1000	1000	1000	1000	4000
Délina	500	500	300	0	1300
Bertha	0	0	1000	100	1100
Moyenne par étudiant	600	875	825	350	2650

FIG. 2 – Exemple de document « .odt » affichant les résultats de l'enquête.

1 Introduction à « ODF » (OpenDocument)

Le format de document OpenDocument, également connu sous le sigle abrégé « ODF », est une norme pour la représentation de données pour les applications de bureautique : Traitements de texte, chiffriers, présentations, diagrammes, dessins et bases de données. L'appellation officielle de la norme est *OASIS Open Document Format for Office Application* et elle est utilisée dans plusieurs applications dont, entre autres, KOffice, OpenOffice, Google Document et Lotus Symphony.

Plusieurs extensions de fichier sont associées à la norme selon le type de fichier utilisé. Celui qui nous intéresse est le texte formaté dont l'extension est « .odt ».

1.1 Les bases du « ODF »

Un document (fichier « .odt ») respectant la norme « ODF » est en fait une archive ZIP contenant un certain nombre de fichiers et de répertoires. Les fichiers et répertoires minimaux dont nous avons besoin sont :

- **content.xml** qui est le fichier qui contient le texte et son formatage en XML.
- **Pictures** qui est le répertoire contenant les images utilisées dans le texte.
- **META-INF** qui est le répertoire contenant les informations sur le fichier **jar**.

Pour les besoins du travail, nous ne touchons qu'au fichier **content.xml**. Les autres fichiers sont fournis et n'ont pas à être modifiés. Pour créer un fichier « .odt », il suffit de créer une archive avec la commande **zip** après avoir récupéré le répertoire **META-INF**

et Pictures, disponibles sur le site WEB et après avoir générer le fichier `content.xml`. La commande exacte pour créer le fichier « `.odt` » est :

```
zip -r nom_fichier.odt content.xml META-INF
```

Ce que vous devez faire est de produire un fichier XML, en l'occurrence le fichier `content.xml`. Produire un fichier XML ressemble à produire un fichier HTML. Cela consiste à mettre en page du texte, des images, des applications en utilisant des commandes: les TAGS ou balises. Ces TAGS sont le plus souvent des commandes pour indiquer la taille des caractères, la couleur du texte, etc... Ces balises se présentent souvent sous la forme: `<commande>` et `</commande>` ou plus rarement sous la forme `<commande/>` .

1.2 La structure du document `content.xml`

Pour générer le document `content.xml`, mieux vaut d'abord en connaître la structure. Le document `content.xml` est composé :

1. d'une balise de version XML et d'encodage (obligatoire):

```
<?xml version="1.0" encoding="us-ascii">
```

2. d'une balise d'entrée :

```
<office:document-content namespaces office:version="1.2">
```

Les « namespaces » donnent les adresses où trouver les descriptions des champs utilisés dans le document. Le fichier `template` disponible sur la page WEB contient la liste de tous les champs à fournir.

Le champs « `<office:version>` » donne la version de « ODF » utilisé. Ce champs doit toujours être présent.

3. de la description des styles utilisés entre les balises (optionnelles) :

```
<office:automatic-styles> et </office:automatic-styles>
```

Dans cette section, il faut définir les styles utilisés dans le document.

4. d'un corps compris entre les balises `<office:body>` et `</office:body>`

Le corps du document, dans notre cas particulier contient du texte. Le texte est encadré par les balises `<office:text>` et `</office:text>` . Dans la section texte, nous pouvons mettre :

- des paragraphes : `<text:p attributs> texte </text:p>` ,
- des tables `<table:table attributs> description et contenu </table:table>`
- des images `<draw:frame attributs> insertion de l'image </draw:frame>`

5. d'une balise de sortie: `</office:document-content>`

Voyons un premier exemple simple de document « ODF ». La figure 3 montre un document « ODF » contenant du texte formaté (« `.odt` »). Dans cet exemple, la ligne (1) spécifie la version de XML utilisé ainsi que l'encodage au format « `utf8` ». Les lignes (2) à (5) contiennent la balise d'ouverture. Les lignes (3) et (4), de la balise d'ouverture, contiennent les

« namespaces ». Ceux-ci indiquent que le document utilise les espaces « **office** » et « **text** ». La ligne (5) donne la version de « ODF » utilisée. La ligne (6) ouvre le corps du document où l'on peut mettre du texte, des figures, un chiffrier, ... La ligne (7) ouvre la section du document contenant du texte. La ligne (8) décrit une entête avec le texte associé. La ligne (9) décrit un paragraphe de texte. Les lignes (10)-(11)-(12) sont les balises de fermeture. La figure 4 nous donne le résultat d'affichage de ce premier exemple.

```
(1) <?xml version="1.0" encoding="UTF-8"?>
(2) <office:document-content
(3)     xmlns:office="urn:oasis:names:tc:opendocument:xmlns:office:1.0"
(4)     xmlns:text="urn:oasis:names:tc:opendocument:xmlns:text:1.0"
(5)     office:version="1.2">

(6)   <office:body>
(7)     <office:text>

(8)       <text:h> Devinez !</text:h>
(9)       <text:p> Bonjour toute la Terre !</text:p>

(10)     </office:text>
(11)   </office:body>
(12) </office:document-content>
```

FIG. 3 – Exemple 1



FIG. 4 – Affichage de l'exemple.

Il existe d'autres sections ou balises que nous présentons plus loin ou bien qui ne sont pas utiles dans le cadre de ce travail.

1.2.1 Les styles

Si un texte contient du texte, des figures ou des tableaux qui n'utilisent pas les formats de base, on doit spécifier le nouveau format dans des styles. Les styles permettent de spécifier le type et la taille des polices ainsi que certains attributs tels l'alignement, le gras et l'italique.

Toutes les descriptions de styles utilisés dans le document doivent être préalablement définies dans la section de style automatique entre les balises: `<office:automatic-styles>` et `</office:automatic-styles>` .

La définition d'un style a la structure suivante :

```

(1)    <style:style style:name="nom" style:family="table"
(2)                style:parent-style-name="Standard">
(3)    <style:table-properties style:width="12cm" table:align="center"/ >
(3)    </style:style>

```

Aux lignes (1) et (3) on retrouve les balises d'ouverture et de fermeture. La balise d'ouverture contient certains attributs dont :

- **style:name** permet de donner un nom au style. Ce nom sera utilisé dans le texte pour appliquer le style au texte.
- **style:family** sert à reprendre des attributs d'un style existant. Les styles existants que nous utilisons sont :
 - « paragraph » pour définir le style à appliquer sur le texte d'un paragraphe.
 - « table » pour définir le style à appliquer sur une table en entier telle la taille de la table.
 - « table-column » pour définir le style à appliquer sur une colonne d'une table telle la largeur de la colonne.
 - « table-cell » pour définir le style à appliquer sur une cellule d'une table telle la taille de la police et les bordures.
 - « graphic » pour définir le style à appliquer sur une image telle sa taille.
- **style:parent-style-name** sert à indiquer l'utilisation d'un style existant. Les styles qui nous intéressent sont:
 - « Graphics » pour définir le style à appliquer sur une image.
 - « Standards » pour définir le style à appliquer sur du texte.

À l'intérieur des balises, on retrouve la description du style. Dans notre exemple (ligne 2) on définit les attributs d'un tableau : sa largeur (12 cm) et son alignement (centré). Il existe des attributs de style pour le texte, les paragraphes, les tableaux, les colonnes de tableaux, les cellules de tableaux ainsi que les figures.

L'exemple de la figure 5 montre l'utilisation des styles pour les paragraphes et le texte. Le paragraphe est centré et les polices sont en gras et d'une taille de 24 points. La figure 6 nous donne le résultat d'affichage de cet exemple.

Il existe des styles spécifiques pour les tableaux et les images. Nous les présentons en même temps que les commandes pour créer des tableaux et des images.

1.2.2 Les paragraphes

Dans ODF, on peut définir séparément les attributs de chaque paragraphe. Chaque paragraphe de texte doit donc être encadré par les balises `<text:p attributs>` et `</text:p>`. Le seul *attribut* dont nous avons besoin est **text:style-name="s1"** qui permet d'associer un style au texte du paragraphe. L'exemple de la figure 5 montre l'utilisation des styles pour les paragraphes et le texte.

```

<?xml version="1.0" encoding="UTF-8"?>
<office:document-content
  xmlns:office="urn:oasis:names:tc:opendocument:xmlns:office:1.0"
  xmlns:style="urn:oasis:names:tc:opendocument:xmlns:style:1.0"
  xmlns:text="urn:oasis:names:tc:opendocument:xmlns:text:1.0"
  xmlns:fo="urn:oasis:names:tc:opendocument:xmlns:xsl-fo-compatible:1.0"
  xmlns:script="urn:oasis:names:tc:opendocument:xmlns:script:1.0"
  office:version="1.2">

  <office:automatic-styles>
    <style:style style:name="P1" style:family="paragraph"
      style:parent-style-name="Standard">
      <style:paragraph-properties fo:text-align="center"/>
      <style:text-properties fo:font-size="24pt" fo:font-weight="bold"/>
    </style:style>
  </office:automatic-styles>

  <office:body>
    <office:text>

      <text:p text:style-name="P1"/>
      <text:p text:style-name="P1"> Mon premier document ODF </text:p>
      <text:p text:style-name="P1"/>

      <text:p > Ceci est un paragraphe avec style.</text:p>
    </office:text>
  </office:body>
</office:document-content>

```

FIG. 5 – Exemple 2

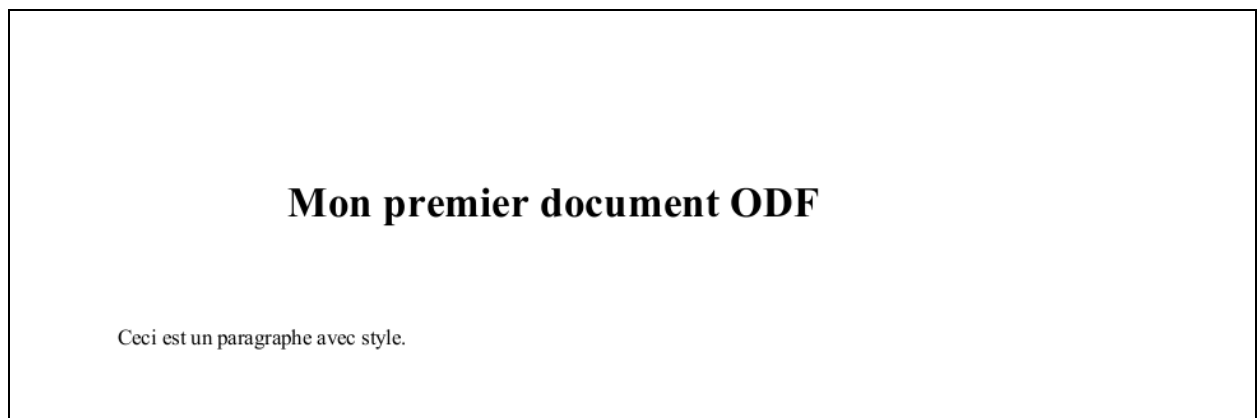


FIG. 6 – Affichage de l'exemple.

1.2.3 Les tableaux

La définition d'un tableau se fait en deux étapes. Rappelons qu'un tableau est composé de lignes, de colonnes et de cellules. La première étape consiste donc à définir les styles pour le tableau, les lignes et les cellules. Pour le tableau, il faut spécifier sa taille et son alignement. Pour les colonnes, il faut spécifier leur largeur. Finalement, pour les cellules, il faut spécifier les bordures et les marges.

La seconde étape consiste à définir le tableau et son contenu en utilisant nos styles. La définition d'un tableau commence par la balise

```
<table:table table:name="nom" table:style-name="style">
```

et termine avec la balise `</table:table>`. Entre ces balise, on retrouve :

1. la balise

```
<table:table-column table:style-name="Col1"
                                table:number-columns-repeated="2"/>
```

qui sert à indiquer le nombre de colonnes dans le tableau.

2. les balises `<table:table-row>` et `</table:table-row>` qui indique le début et la fin d'une ligne.
3. les balises `<table:table-cell table:style-name="Cell1" >` et `</table:table-cell>` qui doivent se situer sur une ligne et encadrent le contenu d'une cellule.
4. le contenu d'une cellule qui est du texte (`<text:p> texte </text:p>`)

L'exemple illustré à la figure 7 montre comment spécifier des styles pour une tableau. La ligne (1) indique que le tableau aura 8 cm de large et qu'il sera centré dans la page. La ligne (2) donne le style pour les colonnes. Les colonnes auront 4 cm de large. La ligne (3) donne le style pour les cellules. Les cellules auront une bordure de 0,5 cm de chaque côté. De plus le texte commencera à 0,2 cm du bord du tableau (padding).

Dans cet exemple, la ligne (4) déclare un tableau centré d'une largeur de 12 cm (style *Tab1*). La ligne (5) indique que le tableau possède 2 colonnes de 4 cm chacune (style *Col1*). Les lignes (6) et (11) encadre une ligne. Les lignes (7) et (9) encadrent une cellule avec des bordures sur ses quatre côtés et une marge de 0,2 cm (style *Cell1*).

La figure 8 montre l'apparence de ce fichier dans OpenOffice.

1.2.4 Les figures

L'insertion d'une image dans le document se fait aussi en deux étapes. La première étape consiste donc à définir les styles pour l'image. Cela concerne surtout comment l'image s'intègre dans le texte du paragraphe courant. La deuxième étape concerne l'insertion de l'image.

L'insertion de l'image débute par la balise

```
<draw:frame draw:style-name="nom" text:anchor-type="paragraph"
            svg:width="10cm" svg:height="2cm">
```



```

<?xml version="1.0" encoding="UTF-8"?>
<office:document-content
  xmlns:office="urn:oasis:names:tc:opendocument:xmlns:office:1.0"
  xmlns:style="urn:oasis:names:tc:opendocument:xmlns:style:1.0"
  xmlns:text="urn:oasis:names:tc:opendocument:xmlns:text:1.0"
  xmlns:fo="urn:oasis:names:tc:opendocument:xmlns:xsl-fo-compatible:1.0"
  xmlns:table="urn:oasis:names:tc:opendocument:xmlns:table:1.0"
  office:version="1.2">

<office:automatic-styles>
(1) <style:style style:name="Tab1" style:family="table">
    <style:table-properties style:width="8cm" table:align="center"/>
  </style:style>
(2) <style:style style:name="Col1" style:family="table-column">
    <style:table-column-properties style:column-width="4cm" />
  </style:style>
(3) <style:style style:name="Cell1" style:family="table-cell">
    <style:table-cell-properties fo:padding="0.2cm"
      fo:border-left="0.05cm solid" fo:border-right="0.05cm solid"
      fo:border-top="0.05cm solid" fo:border-bottom="0.05cm solid"/>
  </style:style>
</office:automatic-styles>

<office:body>
  <office:text>

(4)   <table:table table:name="Table1" table:style-name="Tab1">
(5)     <table:table-column table:style-name="Col1"
        table:number-columns-repeated="2"/>
(6)       <table:table-row>
(7)         <table:table-cell table:style-name="Cell1" >
(8)           <text:p> test1 </text:p>
(9)         </table:table-cell>
(10)        <table:table-cell table:style-name="Cell1" >
            <text:p > 4 </text:p>
          </table:table-cell>
(11)      </table:table-row>
          <table:table-row>
            <table:table-cell table:style-name="Cell1" >
              <text:p> 5 </text:p>
            </table:table-cell>
            <table:table-cell table:style-name="Cell1" >
              <text:p> 7 </text:p>
            </table:table-cell>
          </table:table-row>
        </table:table>

    </office:text>
  </office:body>
</office:document-content>

```

FIG. 7 – Exemple 3

test1	4
5	7

FIG. 8 – Affichage de l'exemple.

et termine avec la balise `</draw:frame>` .

Les attributs dans la balise d'ouverture indique la taille de l'image.

Entre ces balise, on retrouve la balise

`<draw:image xlink:href="Pictures/logo.jpg"/>`

qui sert à donner le nom du fichier qui contient l'image à intégrer dans le texte.

L'exemple de la figure 9 montre comment spécifier un style pour une image. La ligne (1) de l'exemple indique que l'image est de la famille des graphiques et qu'elle ne chevauchera pas le texte (« `style:wrap="none"` »).

Les lignes (2) et (4) déclare une image de 10 cm par 2cm qui ne s'intégrera pas dans un paragraphe (style *fr1* avec le champs `warp=none`). La ligne (3) indique que l'image se trouve dans le fichier `logo.jpg`.

La figure 10 montre l'apparence de ce fichier dans OpenOffice.

1.2.5 Exemple complet

La figure 9 montre un exemple complet comprenant du texte, une image et un tableau. La figure 12 montre l'apparence de ce fichier dans OpenOffice.

Informations supplémentaires

Pour des descriptions plus complètes des documents « ODF », consultez les sites webs suivants:

- <http://fr.wikipedia.org/wiki/OpenDocument>
- <http://en.wikipedia.org/wiki/OpenDocument>
- <http://books.evc-cit.info/>
- http://www.oasis-open.org/committees/documents.php?wg_abbrev=office
- <http://blogs.codes-sources.com/neodante/archive/2008/10/16/opendocument-let-s-go-for-the-first-post-introduction.aspx>

```

<?xml version="1.0" encoding="UTF-8"?>
<office:document-content
  xmlns:office="urn:oasis:names:tc:opendocument:xmlns:office:1.0"
  xmlns:style="urn:oasis:names:tc:opendocument:xmlns:style:1.0"
  xmlns:draw="urn:oasis:names:tc:opendocument:xmlns:drawing:1.0"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:svg="urn:oasis:names:tc:opendocument:xmlns:svg-compatible:1.0"
  office:version="1.2">
  <office:automatic-styles>
    (1) <style:style style:name="fr1" style:family="graphic" style:parent-style-name="Graphics" >
      <style:graphic-properties style:wrap="none" />
    </style:style>
  </office:automatic-styles>
  <office:body>
    <office:text>
      (2) <draw:frame draw:style-name="fr1" text:anchor-type="paragraph"
        svg:width="10cm" svg:height="2cm">
      (3) <draw:image xlink:href="Pictures/logo.jpg"/>
      (4) </draw:frame>
    </office:text>
  </office:body>
</office:document-content>

```

FIG. 9 – Exemple 4



FIG. 10 – Affichage de l'exemple.

```

<?xml version="1.0" encoding="UTF-8"?>
<office:document-content
  xmlns:office="urn:oasis:names:tc:opendocument:xmlns:office:1.0"
  xmlns:style="urn:oasis:names:tc:opendocument:xmlns:style:1.0"
  xmlns:text="urn:oasis:names:tc:opendocument:xmlns:text:1.0"
  xmlns:table="urn:oasis:names:tc:opendocument:xmlns:table:1.0"
  xmlns:draw="urn:oasis:names:tc:opendocument:xmlns:drawing:1.0"
  xmlns:fo="urn:oasis:names:tc:opendocument:xmlns:xsl-fo-compatible:1.0"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:svg="urn:oasis:names:tc:opendocument:xmlns:svg-compatible:1.0"
  office:version="1.2">

<office:automatic-styles>
  <style:style style:name="Table1" style:family="table">
    <style:table-properties style:width="12cm" table:align="center"/>
  </style:style>
  <style:style style:name="col1" style:family="table-column">
    <style:table-column-properties style:column-width="4cm" />
  </style:style>
  <style:style style:name="cell1" style:family="table-cell">
    <style:table-cell-properties fo:padding="0.2cm" fo:border-left="0.05cm solid"
      fo:border-right="0.05cm solid" fo:border-top="0.05cm solid" fo:border-bottom="0.05cm solid"/>
  </style:style>
  <style:style style:name="fr1" style:family="graphic" style:parent-style-name="Graphics">
    <style:graphic-properties style:wrap="none" />
  </style:style>
  <style:style style:name="P2" style:family="paragraph" style:parent-style-name="Standard">
    <style:paragraph-properties fo:text-align="center"/>
    <style:text-properties fo:font-size="14pt" fo:font-weight="bold"/>
  </style:style>
  <style:style style:name="P1" style:family="paragraph" style:parent-style-name="Standard">
    <style:paragraph-properties fo:text-align="center" />
    <style:text-properties fo:font-size="24pt" fo:font-weight="bold"/>
  </style:style>
</office:automatic-styles>

<office:body>
  <office:text>

    <draw:frame draw:style-name="fr1" text:anchor-type="paragraph" svg:width="10cm" svg:height="2cm">
      <draw:image xlink:href="Pictures/logo.jpg"/>
    </draw:frame>

    <text:h>Ceci est un test assez long d'une entête</text:h>
    <text:p text:style-name="P1"/>
    <text:p text:style-name="P1">Ceci est un test de paragraphe </text:p>
    <text:p text:style-name="P1">Ceci est un test de paragraphe </text:p>
    <text:p text:style-name="P1"/>

    <table:table table:name="Table1" table:style-name="Table1">
      <table:table-column table:style-name="col1" table:number-columns-repeated="2"/>
      <table:table-row>
        <table:table-cell table:style-name="cell1" >
          <text:p text:style-name="P2"> test1 </text:p>
        </table:table-cell>
        <table:table-cell table:style-name="cell1" >
          <text:p>4 </text:p>
        </table:table-cell>
      </table:table-row>
      <table:table-row>
        <table:table-cell table:style-name="cell1" >
          <text:p>5 </text:p>
        </table:table-cell>
        <table:table-cell table:style-name="cell1" >
          <text:p>7 </text:p>
        </table:table-cell>
      </table:table-row>
    </table:table>
  </office:text>
</office:body>
</office:document-content>

```

FIG. 11 – Exemple 5



Ceci est un test de paragraphe assez long
Ceci est un test de paragraphe assez long

test1	4
5	7

FIG. 12 – Affichage de l'exemple.