Class BindingTable

{

Name, Alias, (DataTable) Table, KeyColumn, List<BindingField>

}

Class BindingField

{

Name, Alias, (DataType) type, (DataColumn) column, string FormatString

}

Class Include

{

}

Public class BindingKey

{

Public BindingTable Table;

Public Guid Key;  
}

List<BindingKey)> contextList;

Replace(Document document, BindingKey key = null)

{

If (key != null) contextList.Add((table, key));

For fields // text

For tables & includes // comment

for all {annotations} in document.docx // field, include, table

{

If (annotation.Type== field)

{

ReplaceText();

}

If (annotation.Type==include)

{

Table table = GetTable(annotation.TableAlias);

For all (row in table)

{

id = row.ID;

Replace(annotation.Document, key:(table, id);

}

InsertDocument();

}

If (annotation.Type == table)

{

Row[] templateRows = getTemplateRows(DocumentTable);

For all (row in table)

{

id = row.ID;

Row[] rows = replicateRows(templateRows);

Replace(row, key:(table, id);

}

}

}

If (key != null) contextList.Remove(contextList.Count-1);

}

Replace(Row row, BindingKey key = null)

{

If (key != null) contextList.Add((table, key));

for all {annotations} in row

{

If (annotation.Type== field)

{

ReplaceText();

}

}

If (key != null) contextList.Remove(contextList.Count-1);

}

// annotation = {type, filename, tableAlias}