Документация по коду Power Query

# 1. Введение

Данный документ содержит описание кода, использующегося для обработки и объединения данных из различных источников. Код применяется для работы с таблицами, включая очистку, обработку дубликатов и объединение данных. Ниже представлена документация по основным функциям и шагам работы с кодом.

# 2. Первоначальные шаги

1. Открыть книгу "Формирование данных из ОЛ".  
2. На листе "файлы" встать на ячейку B1 и вставить путь к папке с ОЛ.  
3. Проверить все опросные листы на состав столбцов.  
4. Проверить соответствие данных столбцов для всех листов.  
5. При необходимости откорректировать структуру данных для их корректного объединения.

6. Выбрать вкладку «Данные», «Обновить все»

# 3. Описание функций

## 3.1. Функция "ЯвляетсяПустым"

Эта функция проверяет, является ли значение пустым. Входной параметр - это значение любого типа данных.   
 Функция возвращает "true", если значение равно null или является пустой строкой, в противном случае возвращает "false".  
 Код:  
 ```  
 ЯвляетсяПустым = (значение as any) as logical =>  
 let  
 результат = значение = null or (Value.Is(значение, type text) and значение = "")  
 in  
 результат  
 ```

## 3.2. Функция "ОбработкаДубликатов"

Эта функция обрабатывает дубликаты данных в таблице.   
 Она добавляет новый столбец для сцепки ФИО и учетной записи, затем группирует таблицу по этому столбцу.  
 После группировки она проверяет количество строк в группе и объединяет их данные, игнорируя пустые значения.  
 Код:  
 ```  
 ОбработкаДубликатов = (ВходнаяТаблица as table, Text as text) as table =>  
 let  
 ДобавленаСцепка = Table.AddColumn(  
 ВходнаяТаблица,  
 Text & "\_Сцепка\_ФИО\_УЗ",  
 each Text.Combine({  
 Text.From(Record.FieldOrDefault(\_, "Учетная запись", "")),  
 Text.From(Record.FieldOrDefault(\_, "ФИО", ""))  
 }, "|"),  
 type text  
 ),  
 Группировка = Table.Group(  
 ДобавленаСцепка,  
 {Text & "\_Сцепка\_ФИО\_УЗ"},  
 {{"ВсеСтроки", each \_, type table}}  
 ),  
 Обработанные = List.Transform(Группировка[ВсеСтроки], (строки) =>  
 let  
 Количество = Table.RowCount(строки),  
 Результат = if Количество = 1 then строки{0} else  
 let  
 Первая = строки{0},  
 Остальные = List.Skip(Table.ToRecords(строки), 1),  
 ВсеПоля = Record.FieldNames(Первая),  
 Объединённая = Record.FromList(  
 List.Transform(ВсеПоля, (поле) =>  
 let  
 ЗначениеОригинала = try Record.Field(Первая, поле) otherwise null,  
 ЗначенияИзОстальных = List.Transform(Остальные, each try Record.Field(\_, поле) otherwise null),  
 ВсеЗначения = List.InsertRange(ЗначенияИзОстальных, 0, {ЗначениеОригинала}),  
 Отфильтрованные = List.Select(ВсеЗначения, each not ЯвляетсяПустым(\_)),  
 ПервоеЗаполненное = List.First(Отфильтрованные, null)  
 in  
 ПервоеЗаполненное  
 ),  
 ВсеПоля  
 )  
 in  
 Объединённая  
 in  
 Результат  
 ),  
 КакТаблица = Table.FromRecords(Обработанные),  
 БезСцепки = Table.RemoveColumns(КакТаблица, {Text & "\_Сцепка\_ФИО\_УЗ"})  
 in  
 БезСцепки  
 ```

## 3.3. Функция "ПереименованиеСтолбцов"

Эта функция переименовывает столбцы в таблице, чтобы привести их к единому формату. Она выполняет замену некоторых имен столбцов.  
 Код:  
 ```  
 ПереименованиеСтолбцов = (таблица as table) as table =>  
 let  
 результат = Table.RenameColumns(  
 таблица,  
 {  
 {"Учетная запись в MS AD", "Учетная запись"},  
 {"Учетная запись MS AD", "Учетная запись"},  
 {"Учетная запись в службе каталогов", "Учетная запись"},  
 {"Электронная почта\*", "Электронная почта"},  
 {"Мобильный телефон\*", "Мобильный телефон"}  
 },  
 MissingField.Ignore  
 )  
 in  
 результат  
 ```

## 3.4. Функция "ОбработатьПодключаемую"

Эта функция обрабатывает подключаемые таблицы, применяя переименование столбцов, фильтрацию столбцов и обработку дубликатов перед объединением.  
 Код:  
 ```  
 ОбработатьПодключаемую = (  
 ТекущаяТаблица as table,  
 Подключаемая as table,  
 УдаляемыеСтолбцы as list,  
 Модуль as text  
 ) as table =>  
 let  
 ЕстьПодключаемая = Подключаемая <> null,  
 Переименованная = if ЕстьПодключаемая then ПереименованиеСтолбцов(Подключаемая) else null,  
 НазванияСтолбцов = if ЕстьПодключаемая then Table.ColumnNames(Переименованная) else {},  
 СтолбцыКОставлению = if ЕстьПодключаемая then List.Difference(НазванияСтолбцов, УдаляемыеСтолбцы) else {},  
 УдалениеПередОбъединением = if ЕстьПодключаемая then Table.SelectColumns(Переименованная, СтолбцыКОставлению) else null,  
 Объединение = if ЕстьПодключаемая then Table.Combine({ТекущаяТаблица, УдалениеПередОбъединением}) else ТекущаяТаблица,  
 Результат = if ЕстьПодключаемая then ОбработкаДубликатов(Объединение, Модуль) else Объединение  
 in  
 Результат  
 ```

## 3.5. Финальная фильтрация

Этот шаг выполняет фильтрацию строк в таблице, удаляя те, в которых значения в столбцах "ФИО" и "Учетная запись" являются пустыми.  
 Код:  
 ```  
 УдалениеПустыхСтрок = Table.SelectRows(  
 Объединённая,  
 each not ЯвляетсяПустым(Record.FieldOrDefault(\_, "ФИО", null))  
 and not ЯвляетсяПустым(Record.FieldOrDefault(\_, "Учетная запись", null))  
 )  
 ```