Інструкція по СОМ-об'єктам (інтеграція з обліковими системами)

Актуально починаючи з версії М.Е.Doc 11.02.099

Зміст	
Вступ	5
1. Опис документів та VBScript програми документа	6
1.1. Структура документа	6
1.2. Імена полів у документі	6
1.3. VBScript – програми документа	9
Приклади використання вкладки шаблону Програми	10
«CREATE»	11
«OPEN»	12
«SAVE»	12
«MODIFY»	12
«BEFORE_SPR»	13
«AFTER_SPR»	13
2. Робота з документами	14
2.1. Створення документів	15
Відкриття або створення документа розділу "Реєстр звітів" та ПН/РК по ExDocID	15
Створення документу розділу "Реєстр звітів" та ПН/РК	16
Приєднання дочірнього документа до головного (для пакетних звітів)	17
Відкриття або створення документу розділу "Реєстр електронних документів"	18
Створення "Реєстру виданих та отриманих податкових накладних"	19
Перелік RSTTYPE:	20
Перелік ZPeriodTypEnum:	21
Отримати максимальний номер існуючої порції РВтОПН в М.Е.Doc	21
Створення інформаційної довідки «Запит щодо отримання відомостей з ЄРПН»	22
Створення Заявки на формування кваліфікованого сертифіката підписувача ЕП	22
Перелік ZCertReqTypes:	22
2.2. Робота з вмістом документа	24
Отримати документ по CardCode - ідентифікатору документа в БД М.Е.Doc	24
Отримати документ по ExDocID - зовнішньому ідентифікатору документа	24
Отримання списку документів за датою і номером документу	25
Отримання списку документів за типом docType	26
Перелік типів документа docType As Long:	27
Перелік Sender As Long:	28
Перелік Status As Long:	28

Отримання списку документів за розширеними критеріями пошу	ку 28
Додавання вкладення до первинного документа	29
Видалення не підписаного вкладення до первинного документа	29
Отримати документ та його вкладення у форматі PDF	30
Отримати підписане вкладення або графічне відображення Пер	винного документа 30
Отримати підписаний документ	31
Отримання підписантів документа	31
Отримання списку операцій та квитанцій по документу	32
Перелік OperType As Long:	33
Отримання протоколу руху документа	34
Отримання статусу руху документа по ExDocID	35
Отримання статусу руху документа по CardCode	35
Переміщення документа у корзину по ExDocID	35
Переміщення документа у корзину по CardCode	36
Перевірка наявності документа у корзині по ExDocID	36
Перевірка наявності документа у корзині по CardCode	36
Отримання статусу документа по ExDocID	37
Перелік статусу документа CARD.STATUS As Long:	37
Отримання статусу документа по CardCode	37
Отримати дату створення документа в М.Е.Doc	38
Отримати реєстраційний номер документа ПН/РК у ДПС	38
Отримати дату реєстрації документа ПН/РК у ДПС	38
Отримання Витягу з ЄРПН F/J1400107	39
Отримати останній Витяг з ЄРПН за фільтром даних документа	у Витягу 39
Отримання списку Витягів F/J1400107 в заданому діапазоні	40
Отримання статусу реєстрації ПН/РК в ДПС	41
2.3. Відправка, погодження/відхилення документів	42
Помістити в список до відправки документ по ExDocID	42
Помістити в список до відправки документ по DocID	42
Помістити в список для погодження/відхилення документа	43
Очистити список документів до відправки	43
Виконати відправку документів/повідомлень зі сформованого сп	иску 43
2.4. Довідники	44
Отримання інформації з Довідника	44
Перевірка доступу до довідника тільки на читання(без редагува	ння) 45

Отримати значення генератора внутрішнього ID запису в довіднику	45
Видалити всі записи в довіднику	45
Зберегти зміни в довіднику	46
2.5. Шаблони Первинних документів	47
Отримання списку шаблонів первинних документів	47
Отримання XML опису полів шаблона	47
2.6. Додаткові методи	48
Отримати версію М.Е.Doc	48
Отримати ідентифікатор організації в М.Е.Doc	49
Перехід у вибрану організацію	49
Отримати ЄДРПОУ та код філіалу відкритої організації в М.Е.Doc	50
Отримати список документів за умовою	50
Отримати повідомлення	51
Запис у кеш програми сертифіката, секретного ключа та пароля	52
Перевірка платників ПДВ	52
Отримати посилання на документ	53
2.7. Взаємодія з інтерфейсом М.Е.Doc	54
Відкрити документ в інтерфейсі М.Е.Doc за ExDocID	54
Відкрити документ в інтерфейсі M.E.Doc за CardCode	54
Запуск модуля М.Е.Doc	55
Отримати фільтри відкритого модулю	55
Програмно «Оновити» вікно з переліком документів у М.Е.Doc	56
2.8. IZDocument - об'єкт документа	56
Отримати об'єкт таблиці даних документа	56
Встановити тип документа	57
Зберегти документ	57
Отримати CardCode ідентифікатор документа в БД М.Е.Doc	57
Ідентифікатор головного документу	58
Відключити автоматичний розрахунок документа	58
Включити автоматичний розрахунок документа	59
2.9. IZDataSet - об'єкт таблиця	59
Отримати об'єкт колекції полів поточного рядка таблиці	59
Додати запис в таблицю	59
Редагувати таблицю	59
Номер позиції поточного рядка в таблиці	59

Кількість записів у таблиці	59
Встановити курсор на перший запис таблиці	60
Перехід на наступний запис таблиці	60
Перехід на попередній запис таблиці	60
Перехід на останній запис таблиці	60
Ознака знаходження курсора поза межами початку таблиці	60
Ознака знаходження курсора поза межами кінця таблиці	60
Видалити поточний запис таблиці	61
Очистити таблицю	61
Отримати значення поля поточного запису. Повертає 0 якщо значення Null	61
Отримати значення поля поточного запису. Повертає Null якщо значення Null	61
Встановити/отримати поточне сортування записів таблиці	62
Встановити/отримати значення фільтру відображення даних в таблиці	62
Створити порожній об'єкт (таблицю) IZDataset	62
2.10. IZFields - об'єкт колекція полів рядка таблиці	63
Отримати об'єкт поля запису таблиці	63
Кількість полів у запису таблиці	63
Додавання нового поля в колекцію полів IZDataset	63
Перелік ZDataTypeEnum - типи даних поля:	63
2.11. IZField - об'єкт поля рядка таблиці	64
Назва поля	64
Отримати/встановити значення поля рядка таблиці	64
Тип даних поля	64
2.12. Додаткова інформація	64
Запуск М.Е.Doc з командної строки	64
Отримання ідентифікатора документа ExDocID в 1C	65
Вказівник	66



Інструкція призначається для спеціалістів з автоматизації підприємств, що використовують у своїй роботі **1С** (**7.7**, **8.2**, **8.3**) або іншу систему. Інструкція містить опис структури документів та методи, що використовуються у програмі **М.Е.Doc** для створення та опрацювання документів.

У інструкції також наводяться приклади використання методів програми **М.Е.Doc**, що можуть бути корисні під час роботи. У прикладах використовується мова **1C** та **VBScript**.

1. Опис документів та VBScript програми документа

1.1. Структура документа



Документ складається із: тіла документа (**MAIN,0**), підтаблиць (**TABn,0**) та підсумкових рядків для підтаблиць (**TABn,1**).

Будь-які дані (тіло документа, підтаблиці та підсумки підтаблиць, вибірки, таблиці БД тощо) представлені у системі об'єктом **MEDOC.IZDataSet**. Весь набір операцій цього об'єкту може бути застосованим до будь-якої таблиці.

Виключенням є тіло документа та підсумки по підтаблиці документа, які містять лише 1 запис.

1.2. Імена полів у документі

Спосіб, у який можна переглянути імена полів у документі, залежить від типу документа:

Перегляд імен полів для будь-яких документів, крім первинних документів, що розроблені за власною структурою (шаблоном).

Щоб переглянути імена полів таких документів:

1. Оберіть розділ програми М.Е. Довідники – Реєстр бланків.

- 2. Знайдіть потрібний бланк у реєстрі та відкрийте бланк подвійним кліком мишею.
- 3. Ідентифікатори полів позначені абревіатурами, що містять латинські символи. Наприклад, код бланка **S0301011**, назва: **1-ПВ. Звіт з праці (місячна)**:

Файл Правка Вигляд Сервіс Довідка		
Половне меню Реєстр бланків × Перегляд бланка 50301011 ×		
Розділ І. Кількість працівників, робочий час і фон	д оплати г	іраці
	Код	За звітний
Назва показників	рядка	місяць
Α	Б	1
Фонд оплати праці усіх працівників, тис.грн. (з одним десятковим знаком)	1020	A1020
Сума податку з доходів фізичних осіб, що відрахована з фонду		
оплати праці усіх працівників (крім тимчасової непрацездатності), тис. грн.	1020	41020
(з ооним оесятковим знаком) Сереник ооблікова кількість штатиму працівників, осіб (у цілих нислах)	1030	A1030
Кількість відпрацьованого робочого часу штатними працівниками, под год	1040	71040
(у цілих числах)	1060	A1060
Фонд оплати праці штатних працівників, тис.грн. (з одним десятковим знаком)		
(із ряд. 1020)	1070	A1070
Пояснення до розділу І		
Пояснення до розділу І	10 N	

Для документів, які доступні у інших розділах програми:

- 1. Оберіть розділ програми **М.Е.Doc**, у якому знаходиться документ, наприклад, **Реєстр звітів** або **Первинні документи**.
- 2. Відкрийте документ подвійним кліком мишею та оберіть пункт меню **Правка – Електронна форма** (або натисніть клавішу **<F11>**).
- 3. Відкриється вікно **Електронна форма**. Ліворуч наводяться назви таблиць. Праворуч ідентифікатор поля (у квадратних дужках) та його назва:

Електронна форма		_ · •	×
Таблиці документа			
Головна таблиця - [MAIN]	[А110] - Всього по розділу обсяг продажу без ПДВ		
Таолиця - [ТАВТ]	[А111] - Загальна сума коштів		
	[А17] - Всього по розділу обсяг продажу без ПДВ (20%)		
	[А18] - Всього по розділу обсяг продажу без ПДВ (0%)		
	[А19] - Всього по розділу обсяг продажу без ПДВ (0%)експорт		
	[A2_10] - ТТВ-Обсяг продажу без ПДВ		
	[А2_11] - ТТВ-Загальна сума коштів		
	[А2_4] - ТТВ-одиниця виміру товару		
	[А2_5] - ТТВ-Кількість		
	[А2_6] - ТТВ-Ціна продажу		
	[A2_7] - TTB-Обсяг продажу без ПДВ (20%)		
	[А2_8] - ТТВ-Обсяг продажу без ПДВ (0%)		
	[А2_9] - ТТВ-Обсяг продажу без ПДВ (0%)експорт		
	[АЗ_11] - ЗЗТ-загальна сума коштів		
	[А4_10] - Надано (знижка)-Обсяг продажу без ПДВ		
T-6	[А4_101] - Надано (знижка)-Обсяг продажу без ПДВ		
таолиці документа	[А4_11] - Надано (знижка)-Загальна сума коштів		+
	1	F.	

Також імена полів можна переглянути, натиснувши кнопку панелі інструментів **Показати бланк** (або, обравши пункт меню **Файл – Показати бланк**). Приклад **Податкової накладної**:



Відкриється шаблон документа, що містить ідентифікатори полів:

ос насочит в домыт Головне меню Реєстр пе	🧧 🖤 🤝 🌘 рвинних документ	 ів ×)1201009 × 	Перегляд бланка Ј1	201009 ×
Зведена податкова накладна Складена на операції, звільнені від оподаткування Не підляга наданняю отримувачу (покупцю) з прачиви (заяначаєть ся відповідняй тип прачиви)	N25 N26 N13 N14 N14	Податкові зобов'язання* Податковий кредит* Кас. метод* N24	PZOB PKRE D	
		ПОДАТКОВА	накладна [N11 Дата складані
Постачальник (продавець)		ПОДАТКОВА	накладна	N11 Дага складані Отрі
Постачальник (продавець) ГЛА ГІРМ	_NAME	ПОДАТКОВА	накладна	N11 Дата складан Отр
Постачальник (продавець) (найменувания, пріввидь ім'я, по бя	 Такові - для фізичної (ПОДАТКОВА	накладна	N11 Дата складан

- Перегляд імен полів документів, що розроблені за власною структурою (шаблоном)
 - 1. Оберіть розділ програми **М.Е.Doc Первинні документи Шаблони первинних документів**.
 - 2. У вікні ліворуч оберіть назву шаблона документа та відкрийте шаблон подвійним кліком мишею.
 - У вікні редактора шаблонів імена полів, що можуть приймати певні значення, позначені ідентифікаторами у квадратних дужках.
 Властивості поля відкриваються кліком на певному полі.

🔑 Конструктор шаблона [1C82ABP] - Акт (ціна з ПДВ) *				_ 🗆
Файл Правка Вигляд Вставити Формат Таблиця				
J (Структура шаблона 🛛 🗣	Властивості поля		⁴ с
	🕾 Створити поле	Найменування	Номер докум	ента 🖞
	🕾 Створити динамічну таблицю	Ідентифікатор	NUM	<u>F</u>
	🔊 Лолати довідник	Тип даних	Строка	• B
	🖘 Лолати поле з "Картки установи"	Розмір		100
		В електронни форми Заборонити редагування	<u> </u>	
ЗАТВЕРДЖУЮ ЗАТВЕРДЖУЮ	и додатитрафічне поле	Порожнє поле	-	•
[FIRM_ROKPOS] [SIDE_DIR_POS] [FIRM_NAME] [SIDE_CD_K]	H»	Не друкувати		3
(IMAGEFIELD2) [IMAGEFIELD5]	🖻 Основні поля 🔺	Підказка		- BPE
[IMAGEFIELD3] [IMAGEFIELD6]	Дата документа Номер документа			РУТ
(SIGNERINFO)	Перший підпис	Розоивати на ячеики Невилиме		
	Печатка установи	Штрих-код		
	Печатка установи (відпо	ПІБ підписанта		Teh
АКТ злачі-приймання робіт (надання послуг)	2-й підпис відправника	Печать PDF		Ta
	3-й підпис відправника	ПІБ підписанта контраген		
Re [Rom] Bid [DocbArL] p.	2-й підпис відправника 2-й підпис отримувача	Заборонити редагування		
Ми, що нижче підписалися, представники Замовника [SIDE_CD_K] [SIDE_OTV_FIO], з одного боку, і представник Виконавця [FIRM_NAME] [VO_NAME], з іншого боку, склали цей акт про те, що на підставі наведених документив: документив: Договір: [DOG_NAME] [DOG_NUM_T] [DOG_NUM] [DOG_DATA_T] [DOG_DATE] [OSN_T] [OSN_NUM_T] [OSN_NOMER] [OSN_DATA_T] [CSN_DATA]	 З-й підпис отримувача 4-й підпис отримувача 4-й підпис отримувача Ім'я власника сертифікату Ім'я сертифікату печатки Ім'я сертифікату печатки (

Примітка:

Щоб мати змогу відправляти документ контрагенту, у шаблоні обов'язково повинно бути поле «SIDE_EDRPOU_K» (ЄДРПОУ контрагента) з групи Контрагент.

1.3. VBScript – програми документа

Системні програми	Опис
CREATE	Створення. Виконується при створенні нового документа
OPEN	Відкриття. Виконується при відкритті документа, що існує
MODIFY	 Модифікація тіла документа. Виконується перерахунок (F5, F6) 1) Виконується при зміненні будь-якого поля тіла документа. 2) Виконується при зміненні сумарних полів підтаблиць. Фактично об'єднує всі інші програми модифікації (TABLE, AFTER_SPR). Можна застосовувати як комплексну програму модифікації, але тоді необхідно робити додаткові перевірки на вміст полів. Рекомендується використовувати свої програми для призначених дій (див. нижче).
TABLE	Модифікація на рівні таблиці . Фактично є скорочену версію програми MODIFY. Рекомендується використовувати при зміненні значень у таблицях (якщо вони є у документі).
BEFORE_SPR	Дії до вибору із довідника . Фактично є скороченою версією програми MODIFY. Рекомендується використовувати при роботі з довідниками.
AFTER_SPR	Дії після вибору із довідника . Фактично є скороченою версією програми MODIFY. Рекомендується використовувати при роботі з довідниками.

Відключення / Включення системних програм.

Фактично це програмне керування авто розрахунком у документі.

Може бути корисним у випадках, коли програмний розрахунок сум не може бути використаний. Наприклад, сума не повинна розраховуватися як ціна помножена на кількість, так як ціну бухгалтер корегує на копійку (для округлення), і потрібно передавати дані у М.Е.Doc «як є».

Детальніше : <u>Відключити автоматичний розрахунок документа</u> та <u>Включити автоматичний</u> <u>розрахунок документа</u>

Примітка: відключаються лише програми модифікації, програма створення Create() таким чином не відключається.

Програми за потреби прописуються на вкладці **Програми** в **Конструкторі шаблона**. Фактично є аналогом вкладки **Модуль** у **1С**, або звичайний кодінг у середовищі **Visual Studio**, однак, у обмеженому обсязі. Наприклад, відсутня можливість викликати інтелектуальну підказку (IntelleSence).

Приклади використання вкладки шаблону Програми

Використання вкладки буде розглянуто на прикладі шаблону:

№ [NUN	M] від [DOCDAT	E]				[CITY]
Mi FIRM_N	и, що нижче підпи АМЕ], з іншого бок	салися, представник Замовника [SIDE_CD_ł ку, склали цей акт про те, що на підставі нав Договір № [DOG_NUM] від [[(], з одного боку, едених документ ЮG_DATE]	і представн ів:	ик Виконавця	1
ыиконаві	цем були виконані	і наступні роботи (надані такі послуги):				
Виконаві №	цем були виконані Код ДКПП	і наступні роботи (надані такі послуги): Найменування робіт, послуг	Кіль-ть	Од.	Ціна без ПДВ	Сума без ПДВ
Nº TAB1 F	цем були виконані Код ДКПП 11 [ТАВ1 F7] [ТАВ	і наступні роботи (надані такі послуги): Найменування робіт, послуг 31 F2]	Кіль-ть [TAB1 F3]	<mark>Од.</mark> [TAB1 F4]	Ціна без ПДВ [TAB1 F5]	Сума без ПДВ ПАВ1 F6
Nº TAB1 F	цем були виконані Код ДКПП 1] [ТАВ1 F7] [ТАВ	і наступні роботи (надані такі послуги): Найменування робіт, послуг 31 F2]	Кіль-ть (TAB1 F3)	Од. [TAB1_F4] Разом:	Ціна без ПДВ [TAB1 F5] [SUMWIT	Сума без ПДВ ПАВ1 F6 НОUTPDV
Виконаві №	цем були виконані Код ДКПП 1] [ТАВ1_F7] [ТАВ	і наступні роботи (надані такі послуги): Найменування робіт, послуг 31 F2]	Кіль-ть [TAB1 F3] Сум	Од. [ТАВ1 F4] Разом: а ПДВ:	Ціна без ПДВ (TAB1 F5) [SUMWIT	Сума без ПДВ ПАВ1 F6 HOUTPDV [SUMPDV]

Нагадуємо, що такі поля як **Дата**, **Номер**, **Сума без ПДВ**, **Загальний ПДВ**, **Сума по документу** – це зарезервовані поля, список яких прийнятий у шаблоні на вкладці **Основні поля** (чорним виділені поля, що використовуються на поточний момент у шаблоні):



«CREATE»

Програма виконується 1 раз – тільки під час створення. Аналізує дату документа (встановлюється поточна дата) та заповнює відповідальну особу.

Примітка:

Оскільки дату можна змінити після створення документа, то аналогічний код рекомендується додати у програму модифікації **MODIFY**.

```
12
     SUB CREATE()
13
         With rsMain
             .Edit
14
15
             If .FldVal("DOCDATE") < DateSerial(2018,1,1) Then</pre>
16
                  .FldVal("VO NAME") = "IBaHOB I.I."
17
                  .FldVal("VO POS") = "директор"
18
19
              Else
                  .FldVal("VO NAME") = "Петров П.П."
20
                  .FldVal("VO POS") = "головний бухгалтер"
21
22
              End If
23
24
              . Post
25
         End With
     END SUB
26
```

«OPEN»

При кожному відкритті документа (початкове створення – теж відкриття) буде виводити повідомлення.

```
1
    SUB OPEN()
 2
        With rsMain
 3
          .Edit
 4
 5
            if Date() < DateSerial(2018,1,1) then
                msgbox ("Сьогоднішня дата менше 01.01.2018")
 6
 7
            else
                msgbox ("Сьогоднішня дата ,більша 01.01.2018")
 8
 9
            end if
10
11
          .Post
12
        End With
13 END SUB
```

«SAVE»

Програма виконується при кожному натисканні кнопки **Зберегти** та перевіряє на пусте значення поля **Номер документа**.

```
12
    SUB SAVE ()
13
        With rsMain
14
            .Edit
15
16
            MsgBox("Вітаємо!" + vbcrlf + "Документ успішно збережений !")
17
18
            If .FldVal("NUM")="" Then
19
                MsgBox("Не вказано номер документу!")
20
            End If
21
22
            .Post
23
        End With
24 END SUB
```

«MODIFY»

Виконує декілька дій:

- розрахунок суми у таблиці (кількість * ціну);
- розрахунок суми «Всього» (як підсумкова сума по стовпцю Сума без ПДВ);
- розрахунок ПДВ (як 20% від суми);
- підсумкова сума (як сума без ПДВ + сума ПДВ);
- відображення текстового значення суми.

```
6
   SUB MODIFY()
7
       With rsMain
8
            .Edit
9
10
            Set rsTab1 = Doc.DataSets("TAB1")
11
                    rsTab1.First
12
                    While Not rsTab1.EOF
13
                    rsTab1.Edit
14
                        rsTab1.FldVal("TAB1_A6") = rsTab1.FldVal("TAB1_A3") * rsTab1.FldVal("TAB1_A5")
15
                    rsTab1.Post
16
                rsTab1.Next
17
            Wend
18
19
            Set rsTab1 = Nothing
20
21
            Set rsTab1Sum = Doc.DataSets("TAB1",1)
22
               rsMain.FldVal("SUMWITHOUTPDV") = rsTab1Sum.FldVal("TAB1 A6")
23
            Set rsTab1Sum = Nothing
24
25
            rsMain.FldVal("SUMPDV") = rsMain.FldVal("SUMWITHOUTPDV") * 0.2
             rsMain.FldVal("DOCSUM") = rsMain.FldVal("SUMWITHOUTPDV") + rsMain.FldVal("SUMPDV")
26
27
             rsMain.FldVal("DOCSUM_TEXT") = .FldVal("DOCSUM")
28
29
            . Post
30
        End With
31 END SUB
32
```

«BEFORE_SPR»

Програма виконується перед використанням довідника.

```
10
    SUB BEFORE SPR()
11
       With rsMain
12
             .Edit
13
14
                MsgBox ("Відкриваємо довідник !")
15
16
             . Post
17
        End With
18
    END SUB
```

«AFTER_SPR»

Програма виконується після вибору значення із довідника.

У прикладі розглянуто підключений довідник **Номенклатури** та вибір із нього назви одиниці виміру за зв'язаним довідником + ціни обраної номенклатури.

```
1
    SUB AFTER SPR()
2
       With rsMain
3
 4
            Set rsTab1 = Doc.DataSets("TAB1")
5
            rsTab1.Edit
 6
            If DocEditedFieldName = "TAB1 OV NAME"
                                                    Then
7
               rsTab1.FldVal("TAB1_OV_NAME") = CurrSprDS.FldVal("NAME")
8
            End If
9
            If DocEditedFieldName = "TAB1 NOMENKLATURA NAME" Then
               rsTab1.FldVal("TAB1_NOMENKLATURA_NAME") = CurrSprDS.FldVal("Name")
10
               If CurrSprDS.FldVal("Price") <>0 Then
11
12
                 rsTab1.FldOrigVal("TAB1_A5") = CurrSprDS.FldOrigVal("Price")
13
               Else
                 rsTab1.FldVal("TAB1_A5") = Null
14
15
               End if
            '---- одиниці виміру
16
17
               If Not isempty (CurrSprDS.Fields.Item("NUMHBUMEASURE")) Then
                 Spr = App.GetSprValue("HBUMEASURE", "NUM", currSprDS.FldVal("NUMHBUMEASURE"), "NAME")
18
                 If Cstr(Spr) <> "" Then
19
                    rsTab1.FldVal("TAB1 OV NAME") = Spr
20
21
                 Else
22
                    rsTab1.FldVal("TAB1 OV NAME") = ""
23
                 End if
24
               End if
25
            1 .....
26
            End If
27
            rsTab1.Post
28
            Set rsTab1 = Nothing
29
30
        End With
31 END SUB
```

2. Робота з документами

Об'єкти для взаємодії з М.Е.Doc

Поточна програма:

MEDOC.ZApplication

Головний об'єкт для взаємодії з М.Е.Doc - який створюється першим. Рекомендується перевірити підключення і обробити виключення при створенні. Без створення цього об'єкту подальша робота неможлива.

VbScript: set App = CreateObject("MEDOC.ZApplication") 1C 7.7: App = СтворитиОб'єкт("MEDOC.ZApplication"); 1C 8.2: App = Новий СОМОб'єкт("MEDOC.ZApplication");

Документ:

MEDOC.IZDocument

Представляє собою інтерфейс документа через який отримується доступ до даних документа. Властивості та методи - описані в розділі : <u>IZDocument - об'єкт документа</u>

Таблиця даних: MEDOC.IZDataSet Представляє собою інтерфейс взаємодії з таблицею даних. Властивості та методи - описані в розділі : <u>IZDataSet - об'єкт таблиця</u>

Колекція полів рядка таблиці:

MEDOC.IZFields

Представляє собою інтерфейс взаємодії з рядком таблиці даних IZDataSet. Властивості та методи - описані в розділі : <u>IZFields - об'єкт колекція полів рядка таблиці</u>

Поле рядка таблиці:

MEDOC.IZField

Представляє собою інтерфейс взаємодії з полем рядка IZFields таблиці IZDataSet. Властивості та методи - описані в розділі : <u>IZField - об'єкт поля рядка таблиці</u>

2.1. Створення документів

Відкриття або створення документа розділу "Реєстр звітів", ПН/РК та інформаційних довідок по ExDocID

OpenOrCreateByExDocID(TmpIName As String, PeriodType As ZPeriodTypEnum, CrtDocDate As Date, PartName, ExDocID As String, reWrite As Boolean) As IZDocument

Повертає документ за вказаними параметрами, якщо документа не існує - то відбувається створення нового документа. Якщо документ існує та вказано reWrite = true - то виконується перезапис існуючого документа.

Вхідні параметри:

TmplName – чаркод бланка;

PeriodType – тип періоду з Перелік <u>ZPeriodTypEnum;</u>

CrtDocDate – дата створення документа(вказується з 01 числом, наприклад : 01.11.2020);

PartName – параметр зарезервовано, вказується порожнім;

ExDocID – зовнішній ідентифікатор документа;

reWrite – ознака виконання повторного створення існуючого документа при співпадінню **ExDocID** - true/false (1/0).

Документ буде створено у організації яка відкрита в М.Е.Doc. Щоб змінити організацію скористайтесь методом <u>SelectFirm()</u>

Відповідь:

Об'єкт документа IZDocument

Приклад:

set App = CreateObject("Medoc.ZApplication")

```
set Doc = App.OpenOrCreateByExDocID("J1201010", 0, "01.11.2020", , "1234", 0)
      set rsMain = Doc.DataSets()
      rsMain.Edit()
      rsMain.Fields.Item("NAKL TYPE").Value = 1 'вихідний документ
      іцт
Приклад створення інформаційних довідок
      set App = CreateObject("Medoc.ZApplication")
      Set TypeLib = CreateObject("Scriptlet.TypeLib")
      exdocid = TypeLib.Guid
      set doc = App.OpenOrCreateByExDocID("J1400406", 0, "25.02.2023", ,exdocid, 1)
      set Main = Doc.DataSets()
      Main.Edit()
      Main.Fields.Item("FIRM_INN").Value = "5287433691"
      Main.Fields.Item("FIRM NAME").Value = "Установа2"
      Main.Fields.Item("FIRM RUK").Value = "Директор3"
      Main.Fields.Item("FIRM RUKUNN").Value = "52843691258"
      Main.Fields.Item("REP_DATE").Value = "25.02.2023"
      Main.Fields.Item("HJ").Value = 1
      Main.Fields.Item("HK").Value = 1
      Main.Fields.Item("HB").Value = 0
      Main.Post()
      set TAB1 = DOC.DataSets(1)
      TAB1.Edit
      TAB1.Fields.Item("TAB1 A1").Value = "1"
      TAB1.Fields.Item("TAB1_A2").Value = "528743692"
      TAB1.Post()
      Doc.Save()
      Doc.Close()
```

Створення документу розділу "Реєстр звітів" та ПН/РК

CreateDocumentByParam(TmplName As String, PeriodType As ZPeriodTypEnum, CrtDocDate As Date PartName) As IZDocument

Вхідні параметри:

 TmplName – чаркод бланка;

 PeriodType – тип перioду з Перелік <u>ZPeriodTypEnum;</u>

 CrtDocDate – дата створення документа(вказується з 01 числом, наприклад : 01.11.2020);

 PartName – параметр зарезервовано, вказується порожнім;

 Документ буде створено у організації яка відкрита в М.Е.Doc. Щоб змінити організацію скористайтесь методом <u>SelectFirm()</u>

Відповідь:

Об'єкт документа IZDocument

Приклад:

```
set App = CreateObject("Medoc.ZApplication")
'Створимо ЄСВ Додаток 4
set Doc = App.CreateDocumentByParam("J3000412", 0, "01.11.2020", 0)
set DsMain = Doc.DataSets(0)
DsMain.Edit()
...заповнення Переліку таблиць звіту
Doc.Save()
Doc.Close()
۱...
set Doc6 = App.CreateDocumentByParam("J3040612", 0, "01.11.2020", 0)
set DsTab1 = Doc6.DataSets(1)
DsTab1.Edit()
DsTab1.First()
...заповнення Таблиці 6
визначення даного документа як дочірнього для попередньо створеного головного Ј3000412
Doc6.IDParent = Doc.CardCode
Doc6.Save(1)
Doc6.Close()
```

Приєднання дочірнього документа до головного (для пакетних звітів) GetPacketDoc().Add(parentExDocID As String, childExDocID As String) As Long

Вхідні параметри:

parentExDocID – зовнішній ідентифікатор головного документа пакету звітів; childExDocID – зовнішній ідентифікатор дочірнього документа пакету звітів.

Відповідь:

Число - код статусу результату приєднання дочірнього документа до головного, можливі значення:

«0» - дочірній документ додано до головного;

- «1» головний документ не є батьківським;
- «2» дочірній документ не визначений для пакету;

«З» - дочірній документ не може бути доданий до вказаного пакету.

Приклад використання в 1С:

App = Новый СОМОбъект("MEDOC.ZApplication");

// створюємо головний документ **PDate = Дата(2020,11,01);** parentID = "parent"; Doc = App.OpenOrCreateByExDocID("J0200121", 0, PDate, , parentID, 0); // заповнюємо документ ... Doc.Save(1); Doc.Close(); // створюємо дочірній документ childID = "child"; Doc = App.OpenOrCreateByExDocID("J0215121", 0, PDate, , childID, 0); // заповнюємо документ ... Doc.Save(1); Doc.Close(); // приєднуємо дочірній документ до головного packetDoc = App.GetPacketDoc(); КодВозврата = packetDoc.Add(parentID, childID); Если КодВозврата = 0 Тогда Сообщить("Дочірній документ додано до головного"); ИначЕесли КодВозврата = 1 Тогда Сообщить("Помилка додавання у пакет: головний документ не є батьківським"); ИначеЕсли КодВозврата = 2 Тогда Сообщить("Помилка додавання у пакет: дочірній документ не визначений для пакету"); Иначе Сообщить("Помилка додавання у пакет: дочірній документ не може бути доданий до вказаного пакету"); КонецЕсли;

Примітка: якщо раніше уже передавались документи з такими ж ІД (основний або додаток) то необхідно помістити їх до корзини або ж присвоїти при створенні новий ІД

Відкриття або створення документу розділу "Реєстр електронних документів"

OpenOrCreateDocument(orgld As Long, **charCode** As String, **num** As String, **docDate** As Date, **ExDocID** As String, **reWrite** As Boolean) As IZDocument

Повертає документ за вказаними параметрами, якщо документа не існує - то вібувається створення нового документа. Якщо документ існує та вказано reWrite = true - то виконується перезапис існуючого документа.

Вхідні параметри:

orgld - внутрішній ідентифікатор організації в М.Е.Doc, в якій необхідно створити документ. Для отримання скористайтесь методом <u>SeekOrg();</u>

charCode – чаркод шаблону первинного документа;

num – номер документа;

docDate – дата документа;

ExDocID – зовнішній ідентифікатор документа;

reWrite – ознака виконання повторного створення існуючого документа при співпадінню **ExDocID** - true/false (1/0).

Відповідь:

Об'єкт документа IZDocument

Приклад:

set App = CreateObject("Medoc.ZApplication")
orgId = App.SeekOrg("11223344", "")
set pd= App.GetPrimaryDocs()
set Doc = pd.OpenOrCreateDocument(orgId ,"1C82ABP", "4", "24.11.2020", "1234", 0)

Приклад заповнення таблиці маршрутів для багатостороннього первинного документа:

set rsRoute = Doc.DataSets("TAB9000")

rsRoute.Edit() rsRoute.Fields.Item("DOCROUTE_POSITION").Value = 0 rsRoute.Fields.Item("DOCROUTE_EDRPOU").Value = "11223344" rsRoute.Fields.Item("DOCROUTE_NAME").Value = "Продавець"

rsRoute.Append()

rsRoute.Edit() rsRoute.Fields.Item("DOCROUTE_POSITION").Value = 1 rsRoute.Fields.Item("DOCROUTE_EDRPOU").Value = "11110000" rsRoute.Fields.Item("DOCROUTE_NAME").Value = "Покупець1"

rsRoute.Append()

rsRoute.Edit() rsRoute.Fields.Item("DOCROUTE_POSITION").Value = 2 rsRoute.Fields.Item("DOCROUTE_EDRPOU").Value = "22220000" rsRoute.Fields.Item("DOCROUTE_NAME").Value = "Покупець2"

Створення "Реєстру виданих та отриманих податкових накладних" CreateNewRnn(TmplName As String, ExDocID As String, PeriodType As ZPeriodTypEnum, CrtDocDate As Date) As IZDocument

Вхідні параметри:

 TmplName – чаркод бланка;

 ExDocID – зовнішній ідентифікатор документа;

 PeriodType – тип періоду з Перелік <u>ZPeriodTypEnum;</u>

 CrtDocDate – дата створення документа(вказується з 01 числом, наприклад : 01.11.2020);

Відповідь:

Об'єкт створеного Документа IZDocument

Приклад використання в 1С:

```
Арр = Новый СОМОбъект("MEDOC.ZApplication");
НомерПорции = Строка(Документ.HNP);
МойПериод = Дата(01, 11, 2020);
Если КодДокумента = "J1201508" Тогда
```

ТипРННКод = 24; // із Перелік <u>RSTTYPE</u>

ТипРНННазвание = "ЗАГАЛЬНИЙ";

ИначеЕсли

//можемо зробити перевірку на інші типи РПН

КонецЕсли;

//перевірка на наявність порції в періоді

МаксПорцияВПериоде = App.GetRnnPortion(0, МойПериод, ТипРННКод); Если МаксПорцияВПериоде = НомерПорции Тогда

Сообщить(ТипРНННазвание + " реєстр з номером порції " + НомерПорции + "у періоді" + МойПериод + "вже існує!");

//у такому випадку можна змінити вміст документа методом OpenOrCreateByExDocID()

Иначе

Сообщить(ТипРНННазвание + " реєстр з номером порції " + НомерПорции + " Не існує! " + Символы.ПС + "Створимо його.");

Doc = App.CreateNewRnn(КодДокумента, "MyExDocId", 0, МойПериод); Main = Doc.DataSets(); Main.Edit(); Main.Fields.Item("RSTTYPE").Value = ТипРННКод; Main.Fields.Item("HNP").Value = НомерПорции;

КонецЕсли;

При заповненні реєстру необхідно перевіряти період та тип реєстру. В залежності від цього записувати значення коду у поле **RSTTYPE**.

Якщо це поля відсутнє у документі, неможливо буде сформувати на його основі Декларацію та Додаток до неї!

Перелік **RSTTYPE**:

до 01 січня 2010:

- 0 ЗАГАЛЬНИЙ;
- 1 Переробного підприємства;
- 2 Сільськогосподарського підприємства;
- 3 Сільськогосподарського товаровиробника;
- 4 Спеціальний режим оподаткування.

з 01 січня 2010:

- 5 ЗАГАЛЬНИЙ;
- 6 Переробного підприємства (пункт 11.21 Закону);
- 7 Спеціальний режим оподаткування;
- 8 Сільськогосподарського підприємства.

з 01 січня 2011:

- 9 ЗАГАЛЬНИЙ;
- 10 Переробного підприємства;
- 11 Скорочений (стаття 209 Кодексу);
- 12 Спеціальний (пункт 209.18 статті 209 Кодексу);

13 - ЗАГАЛЬНИЙ для фізичних осіб.

з 01 березня 2013:

- 14 ЗАГАЛЬНИЙ;
- 15 спеціальний режим оподаткування переробного підприємства;
- 16 спеціальний режим оподаткування діяльності у сфері с/г;
- 17 сільськогосподарське підприємство, яке не обрало спец.режим;
- 18 ЗАГАЛЬНИЙ для фізичних осіб.

з 01 січня 2013:

- 19 ЗАГАЛЬНИЙ;
- 20 спеціальний режим оподаткування переробного підприємства;
- 21 спеціальний режим оподаткування діяльності у сфері с/г;
- 22 сільськогосподарське підприємство, яке не обрало спец.режим;
- 23 ЗАГАЛЬНИЙ для фізичних осіб.

з 01 грудня 2014:

- 24 ЗАГАЛЬНИЙ;
- 25 спеціальний режим оподаткування переробного підприємства;
- 26 спеціальний режим оподаткування діяльності у сфері с/г;
- 27 сільськогосподарське підприємство, яке не обрало спец.режим;
- 28 ЗАГАЛЬНИЙ для фізичних осіб.

Перелік ZPeriodTypEnum:

- 0 місяць. Для первинних документів, завжди вказуйте 0;
- 10 квартал (по даті PerDate визначається, який саме квартал 01.03; 01.06; 01.09; 01.12);
- 20 1 півріччя (по даті PerDate визначається, яке саме півріччя 01.06; 01.12);
- 25 9 місяців;
- 30 рік.

Отримати максимальний номер існуючої порції РВтОПН в М.Е.Doc

GetRnnPortion(PeriodType As ZPeriodTypEnum, PerDate As Date, RstType As Long) As Long

Вхідні параметри:

PeriodType – тип періоду з Перелік <u>ZPeriodTypEnum;</u>

PerDate – дата створення документа(вказується з 01 числом, наприклад : 01.11.2020); **RstType** – значення із Перелік <u>RSTTYPE</u>.

Відповідь:

Число - максимальне значення поля "HNP" серед документів РВтОПН за RstType

Приклад використання в 1С:

App = Новый СОМОбъект("MEDOC.ZApplication");

МойПериод = Дата(01, 11, 2020);

Если КодДокумента = "J1201508" Тогда

ТипРННКод = 24;// RSTTYPE

ТипРНННазвание = "ЗАГАЛЬНИЙ";

ИначеЕсли

//можемо зробити перевірку на інші типи РПН

КонецЕсли;

МаксПорцияВПериоде = App.GetRnnPortion(0, МойПериод, ТипРННКод);

Створення інформаційної довідки «Запит щодо отримання відомостей з ЄРПН»

CreateGovQryByCharCode(charCode As String) As IZDocument

Вхідні параметри:

charCode – чаркод бланка (F/J1300107).

Відповідь:

Об'єкт створеного Документа IZDocument

Приклад:

```
set App = CreateObject("Medoc.ZApplication")
set InfoDoc = App.CreateGovQryByCharCode("J1300107")
rsMain = InfoDoc.DataSets()
rsMain.Edit()
'...заповнення полів документа
InfoDoc.Save(1)
InfoDoc.Close()
```

Створення Заявки на формування кваліфікованого сертифіката підписувача ЕП

CreateOrOpenCertReq(reqType As ZCertReqTypes, isUSC As Boolean, ExDocID As String) As IZDocument

Вхідні параметри:

reqType – тип сертифіката з переліку ZCertReqTypes; isUSC = false, ExDocID – зовнішній ідентифікатор документа;

Відповідь:

Об'єкт створеного Документа IZDocument

Перелік ZCertReqTypes:

- 11 Заявка печатки
- 12 Заявка директора
- 13 Заявка бухгалтера

- 14 Заявка співробітника
- 15 Заявка фізичної особи

Заявку на сертифікат Z0200108 створюється на основі наявного запиту на створення сертифікату CSR в форматі p10. Для цього за допомогою методу **CreateOrOpenCertReq**() необхідно створити документ заявки, заповнити поля і зберегти його.

Після створення і заповнення заявки необхідно також оновити дані для заявки в модулі. Для цього використовується метод:

RefreshCertReq(ExDocID As String)

Вхідні параметри:

ExDocID – зовнішній ідентифікатор документа (заявки на сертифікат);

При заповненні заявки необхідно передати в поля MEM, MEMC вміст CSR запиту на сертифікат підписання та шифрування в форматі p10, N8, N9 – ідентифікатори відкритих ключів ЕЦП та шифрування, відповідно. Також необхідно подбати про відповідність даних, вказаних в заявці Z0200108 і наданих для формування CSR-запиту.

Відправляти заявку на сертифікат в АЦСК необхідно безпосередньо в програмі **М.Е.Doc** в реєстрі заявок на сертифікат.

Приклад створення заявки на сертифікат:

```
set App =CreateObject("Medoc.ZApplication")
pn date = "08.07.2022"
   exdocid = "zayavasert" ' задаємо зовнішній ідентифікатор
   reqType = 12
   set Doc = App.CreateOrOpenCertReq(reqType, 0, exdocid)
   set Main = Doc.DataSets()
   Main.Edit()
    Main.Fields.Item("A2_1").Value
                                             = "Ключ керівника"
    Main.Fields.Item("A3_11").Value
                                                "X"
                                             =
    Main.Fields.Item("A3 4").Value
                                                "123"
                                             =
                                             = "528743691"
    Main.Fields.Item("DOG1").Value
                                           = "15.07.2025"
    Main.Fields.Item("DT_END").Value
                                           = "5287436911DU-22072114568"
    Main.Fields.Item("DT_FILL").Value
                                           = "15.07.2022"
    Main.Fields.Item("DT START").Value
                                            = "Тестовий Директор"
    Main.Fields.Item("FIO POS").Value
                                           = "Тестовий"
    Main.Fields.Item("FIO POS GN").Value
                                             = "Директор"
    Main.Fields.Item("FIO POS SN").Value
    Main.Fields.Item("FIRM ADR").Value
                                            = "01234, КИЇВ, тупик 187, 2/1"
    Main.Fields.Item("FIRM CITY").Value
                                            = "КИЇВ"
    Main.Fields.Item("EDDRU").Value
                                            = "528743691234234"
    Main.Fields.Item("FIRM EDRPOU").Value = "528743691"
    Main.Fields.Item("FIRM EMAILORG").Value = "test9@test.net"
    Main.Fields.Item("FIRM NAME").Value
                                             = "Підприємство"
    Main.Fields.Item("FIRM RUK").Value
                                             = "Тестовий Директор"
    Main.Fields.Item("FIRM ZIP").Value
                                            = "01234"
    Main.Fields.Item("ISKEYINDEVICE").Value = "0"
                                             = "30...F"
    Main.Fields.Item("MEM").Value
                                             = "30...3FC03"
    Main.Fields.Item("MEMC").Value
                                            = "2c1...7c"
    Main.Fields.Item("N8").Value
                                            = "1b...97e2b4"
    Main.Fields.Item("N9").Value
                                           = "5287436911"
    Main.Fields.Item("NUM").Value
                                         = "Вул. Героїв"
    Main.Fields.Item("ORG ADR").Value
    Main.Fields.Item("ORG_CITY").Value
                                           = "КИЇВСЬКА ОБЛАСТЬ/М.КИЇВ"
    Main.Fields.Item("ORG_EDRPOU").Value = "528743691"
```

Main.Fields.Item("ORG EDRPOU VIS").Value	= "528743691"
Main.Fields.Item("ORG EMAIL").Value	<pre>= "test9@test.net"</pre>
Main.Fields.Item("ORG KOATU").Value	= "320000000"
Main.Fields.Item("ORG NAME").Value	= "Підприємство "
Main.Fields.Item("ORG ZIP").Value	= "65412"
Main.Fields.Item("PIB_OSOB").Value	= "Чингачгук Великий Змій"
Main.Fields.Item("POS").Value	= "Директор"
Main.Fields.Item("P PIB").Value	= "Тестовий Директор"
Main.Fields.Item("REP DATE").Value	= "08.07.2022"
Main.Fields.Item("REQTYPE").Value	= CStr(reqType)
Main.Fields.Item("SMARTID").Value	— "0"
Main.Fields.Item("TELORG").Value	= "9852369"
Doc.Save(1)	
Doc.Close()	
App.RefreshCertReq(ExDocID)	
cardcode = Doc.CardCode	

WScript.Echo "Створено Z0200108, CardCode = " & cardcode & " EXDOC_ID = " & exdocid

2.2. Робота з вмістом документа

Отримати документ по CardCode - ідентифікатору документа в БД М.Е.Doc OpenDocumentByCode(CardCode As Long) As IZDocument

Вхідні параметри:

CardCode – внутрішній ідентифікатор документа у БД М.Е.Doc;

Відповідь:

Об'єкт документа IZDocument

Приклад:

set App = CreateObject("Medoc.ZApplication")

set Doc = App.OpenDocumentByCode(1234)

Отримати документ по ExDocID - зовнішньому ідентифікатору документа OpenDocumentByExDocID (ExDocID As String) As IZDocument

Вхідні параметри:

ExDocID – зовнішній ідентифікатор документа;

Відповідь:

Об'єкт Документа IZDocument

Приклад:

set App = CreateObject("Medoc.ZApplication")

set Doc = App.OpenDocumentByExDocID("1234")

Отримання списку документів за датою і номером документу

SeekDocumens(orgld As Long, charCode As String, num As String, docDate As Date) As IZDataset

Вхідні параметри:

orgld - внутрішній ідентифікатор організації в М.Е.Doc, в якій необхідно створити документ. Для отримання скористайтесь методом SeekOrg(); charCode – чаркод шаблону первинного документа;

num – номер документа;

docDate – дата документа;

Відповідь:

Об'єкт таблиця IZDataset з полями:

СОDE – ідентифікатор документа у БД М.Е.Doc;

DOCDATE – дата документа;

NUM – номер документа;

PEDRPOU – ЄДРПОУ контрагента;

PNAME – Назва контрагента;

DOCSUM – підсумок по документу;

CHARCODE – чаркод документа;

Приклад:

```
set App =CreateObject("Medoc.ZApplication")
orgId = App.SeekOrg("528743691", "")
docDate = "22.07.2022"
docNum = "4"
docCharcode = "1C77BH"
set prim = App.GetPrimaryDocs()
set Documents = prim.SeekDocumens(orgld, docCharcode, docNum, docDate)
detailsInfo ="Первинні документи від "& docDate &" №" & docNum & " - " & Documents.MaxRecords &
" шт " & vbCrLf
  i = 0
  Do While i< Documents.Fields.Count
      name = Documents.Fields.Item(i).name
      detailsInfo = detailsInfo & name & vbTab
     i = i + 1
  Loop
  detailsInfo = detailsInfo & vbCrLf
  'деталі документів
  Documents.First()
  i = 1
  Do While Not Documents.EOF
     i = 0
      detailsInfo = detailsInfo & CStr(j) & "."
      Do While i< Documents.Fields.Count
         val = Documents.Fields.Item(i).value
         detailsInfo = detailsInfo & val & vbTab
        i = i + 1
      Loop
      detailsInfo = detailsInfo & vbCrLf
      Documents.Next()
     j = j + 1
  Loop
WScript.Echo (detailsInfo)
```

Отримання списку документів за типом docType

GetPrimaryReestr(docType As Long, docMove As Long, dateBegin As Date, dateEnd As Date) As IZDataset

Вхідні параметри:

docType - тип документа з Перелік <u>docType</u>. «-1» – всі типи документів;

docMove - напрямок руху документа : -1 – всі документи (вхідний та вихідний), 0 – вхідний, 1 – вихідний;

dateBegin - дата початку діапазону вибірки;

dateEnd - дата кінця діапазону вибірки.

Відповідь:

Об'єкт таблиця <u>IZDataset</u> з полями:

SENDER – походження документа - з Перелік <u>Sender</u>;

CHARCODE – чаркод документа;

CARDCODE – ідентифікатор документа у БД М.Е.Doc;

STATUS – статус - з Перелік Status;

* VATSUM – сума ПДВ;

EXDOC_ID – зовнішній ідентифікатор документа;

REGDATE – дата реєстрації;

DOC_ID – внутрішній ідентифікатор документа;

DOC_TYPE – тип документа - з переліку <u>docType</u>;

DOC_MOVE – напрямок документа;

DOC_VD – вид діяльності;

DOC_NUM – номер документа;

REG_NUM – реєстраційний номер;

DOC_DEPT – номер філії;

DOC_DATE – дата документа;

FIRM_EDRPOU – ЄДРПОУ платника;

FIRM_IPN – IПН платника;

PARTNER_EDRPOU – ЄДРПОУ контрагента;

PATRNER_IPN – IПН контрагента;

PARTNER_DEPT – код філії контрагента;

OSOBFIO – відповідальна особа;

SENDUSER – користувач, що відправив первинний документ;

EXPDATE – дата експорту документа;

NOTATION – примітка.

* Для Додадку 2 (docType = 10101) значення VATSUM дорівнює сумі ПДВ 20% (основна ставка), або дорівнює сумі ПДВ за пільговими ставками (7% + 14%) за умови, що Додаток 2 складено лише для товарів і послуг, що мають пільгові ставки ПДВ.

Для вихідних документів також поля:

GETTIME – дата та час отримання документа контрагентом;

APPROVE – інформація з вкладки «Затверджено» про підписанта, хто затвердив документ зі сторони контрагента.

Приклад:

```
set App = CreateObject("Medoc.ZApplication")
set Prim = App.GetPrimaryDocs()
set DsReestr = Prim.GetPrimaryReestr(10100, -1, "01.11.2020", "30.11.2020")
```

```
DsReestr.First()
do While Not DsReestr.Eof
DocCode = DsReestr.Fields.Item("CARDCODE").Value
DocNum = DsReestr.Fields.Item("DOC_NUM").Value
DocRegNum = DsReestr.Fields.Item("REG_NUM").Value
'...
DsReestr.Next()
```

Loop

Перелік типів документа docType As Long:

«10100» - Податкова накладна; «10101» - Додаток 2.

Первинні документи:

«10072» - Товарно-транспортна накладна;

- «10103» Рахунок;
- «10104» Акт;
- «10105» Накладна;
- «10106» Інші Документи;
- «10107» Банківська виписка;
- «10108» Довіреність;
- «10109» Договір;
- «10110» Акт розбіжностей;
- «10111» Документ з вкладенням;
- «10112» Додаткова угода;
- «10113» Специфікація;
- «10136» Акт звірки;
- «10137» Акційна пропозиція.

Акциз:

- «10050» Акцизна накладна;
- «10051» Розрахунок коригування (акциз);
- «10120» Акцизна накладна форми «П»;
- «10121» Розрахунок коригування форми «П»;
- «10130» Акцизна накладна форми «С»;
- «10131» Розрахунок коригування форми «С»;
- «10122» Довідка про розпорядника акцизного складу пального, акцизні склади пального, розташовані на них резервуари пального, витратоміри та рівнеміри;
- «10123» Довідка про зведені за добу підсумкові облікові дані щодо обсягів обігу

(отримання/відпуску) та виробленого спирту етилового та/або обсягу виробленої продукції; «10135» - Довідка про зведені за добу підсумкові облікові дані щодо обсягів обігу

(отримання/відпуску) та залишків спирту на акцизному складі виробника спирту та/або виробника окремих видів продукції;

«10053» - Заявка на поповнення/коригування залишку пального;

«10133» - Заявка на поповнення/коригування спирту етилового.

Перелік Sender As Long:

- «-1» не визначено;
- «1» ЄРПН;
- «2» контрагент;
- «З» контрагент (імпортовано із зовнішньої системи);
- «10» копіювання у РПН;
- «11» контрагент (прийнято з сервера ДФС).

Перелік Status As Long:

- «0» Готується (для первинних документів) ;
- «1» Відправлено в ДФС (для НН/РК);
- «2» Доставлено в ДФС (для НН/РК);
- «З» Відхилено в ДФС (для НН/РК);

«4» - Прийнято в ДФС (для НН/РК) (Інших значень поля «Status» для ПН/РК не передбачено);

- «5» Відправлено контрагенту (для первинних документів);
- «6» Доставлено контрагенту (для первинних документів);
- «7» Затверджено контрагентом (для первинних документів);
- «8» Отриманий (для первинних документів);
- «9» Відхилено контрагентом (для первинних документів).

Отримання списку документів за розширеними критеріями пошуку

GetPrimaryReestrEx(docType As Long, docMove As Long, dateBegin As Date, dateEnd As Date, Params As String) As IZDataset

Вхідні параметри:

docType - тип документа з Перелік <u>docType</u>. «-1» – всі типи документів;

docMove - напрямок руху документа : -1 – всі документи (вхідний та вихідний), 0 – вхідний, 1 – вихідний;

dateBegin - дата початку діапазону вибірки;

dateEnd - дата кінця діапазону вибірки;

Params – додаткові параметри відбору документів, які задаються у вигляді частини SQL запиту для оператора WHERE. В якості параметрів вказуються і значення полів таблиці <u>IZDataset</u> Наприклад,

"DOC_NUM = '142' and DOC_DATE = '15.07.2022' "

Відповідь:

Об'єкт таблиця <u>IZDataset</u> з полями (див. Отримання списку документів за типом docType).

Приклад:

dateBegin = "01.05.2022" dateEnd = "01.06.2022" Filter_query = "Doc_num='" + 124 + "'" Set Docs = Prim.GetPrimaryReestrEx(-1, -1, dateBegin, dateEnd, Filter_query) Info = "Знайдено " & Docs.MaxRecords & " документів"

Додавання вкладення до первинного документа AddAppendix(fullFileName As String, ExDocID As String, CardCode As Long) As Long

Вхідні параметри:

fullFileName – повний шлях до файлу-вкладення; ExDocID – зовнішній ідентифікатор документа; CardCode – внутрішній ідентифікатор документа у БД М.Е.Doc;

Відповідь:

- Число код статусу результату додавання вкладення, можливі значення: «0» успішно;
 - «1» не вказано ідентифікатор документа;
 - «2» немає документа за вказаним ідентифікатором;
 - «3» для x32 обмеження на 50 вкладень, уже в наявності;
 - «4» обмеження на довжину імені файла, більше 50 символів;
 - «5» обмеження на загальний розмір 20Мб;
 - «6» для x32 обмеження на 50 вкладень з новим вкладенням;
 - «7» помилка додавання у базу;
 - «8» інші помилки, відсутність або блокування файла тощо.

Приклад:

- set App = CreateObject("Medoc.ZApplication")
- set prim = App.GetPrimaryDocs()
- resCodeByExdocid = prim.AddAppendix("C:\File.txt", "1234", 0)
- resCodeByCardcode = prim.AddAppendix("C:\File.txt", "", 1234)

Видалення не підписаного вкладення до первинного документа ClearAppendix(CardCode As Long, ExDocID As String) As Boolean

Вхідні параметри:

CardCode – внутрішній ідентифікатор документа у БД М.Е.Doc; **ExDocID** – зовнішній ідентифікатор документа.

Відповідь:

True – видалено; *False* - не видалено.

Приклад:

set App = CreateObject("Medoc.ZApplication") set Prim = App.GetPrimaryDocs() isDeleted = Prim.ClearAppendix(1234, "") 'a6o 3a ExDocID isDeleted = Prim.ClearAppendix(0, "1234")

Отримати документ та його вкладення у форматі PDF PrintDocPDF(CardCode As Long, ExDocID As String)

Метод дозволяє:

- зберегти у форматі PDF Універсальні документи, Первинні документи та вкладення до них, податкові накладні і розрахунки коригування до них (ПН, ПН/РК), акцизні накладні і розрахунки коригування до них (АН, АН/РК);
- всі вкладення до документа та сам документ зберігаються у окремих файлах формату PDF;
- для підписаного документа та вкладення у PDF-файлі відображається факсиміле підписів та інформація про підписантів;
- у разі відсутності вкладень зберігається безпосередньо Універсальний / Первинний документ.

Вхідні параметри:

CardCode – внутрішній ідентифікатор документа у БД М.Е.Doc; **ExDocID** – зовнішній ідентифікатор документа.

Відповідь:

Об'єкт таблиця <u>IZDataset</u> з полями: FILENAME – ім'я файлу; FILEDATABASE64 – строка даних в base64 у форматі PDF

Приклад:

```
set App = CreateObject("Medoc.ZApplication")
set Prim = App.GetPrimaryDocs()
set pdfs = Prim.PrintDocPDF(123456, "")
'a6o 3a ExDocID set pdfs = prim.PrintDocPDF(0, "123456")
pdfs.First()
While Not pdfs.EOF
msgbox pdfs.Fields.Item("FILENAME").Value
msgbox pdfs.Fields.Item("FILEDATABASE64").Value
pdfs.Next()
Wend
```

Отримати підписане вкладення або графічне відображення Первинного документа GetAppendixSign(CardCode As Long, ExDocID As String, typeapp As Boolean, typesign As Boolean) As IZDataset

Вхідні параметри:

CardCode – внутрішній ідентифікатор документа у БД М.Е.Doc; ExDocID – зовнішній ідентифікатор документа; typeapp – «0» - вкладення; «1» - графічне відображення Первинного документа; typesign – тип підпису : «1» - CAdES.

Відповідь:

Об'єкт таблиця <u>IZDataset</u> з полями: FILENAME – ім'я файлу; FILEDATABASE64 – строка даних в base64 у форматі підпису CAdES

Приклад:

set App = CreateObject("Medoc.ZApplication") set Prim = App.GetPrimaryDocs() set DsAppdx = Prim.GetAppendixSign(1234, "", 0, 1) 'a6o 3a ExDocID set DsAppdx = Prim.GetAppendixSign(0, "1234", 0, 1) AppdxName = DsAppdx.Fields.Item("FILENAME").Value AppdxData = DsAppdx.Fields.Item("FILEDATABASE64").Value

Отримати підписаний документ

GetDocSign(CardCode As Long, ExDocID As String, SignType As Long) As String

Метод повертає підписаний документ за ідентифікатором **CardCode** або **ExDocID** в наступних форматах **SignType** :

«1» - XML без підписів;

«2» - XML з підписами;

«З» - XML з підписами, так як він зберігається у М.Е.Doc;

«4» - Графічне відображення документа (підписаний PDF - тільки Первинні документи крім ПН та PK).

Документ повертається у вигляді строки в base64.

Вхідні параметри:

CardCode – внутрішній ідентифікатор документа у М.Е.Doc; ExDocID – зовнішній ідентифікатор документа; SignType – формат підписаного документа.

Відповідь:

Строка - документ в base64.

Приклад:

set App = CreateObject("Medoc.ZApplication")
set DocSign = App.GetDocSigning()
DocByCardcode = DocSign.GetDocSign(123456, "", 1)
DocByExDocID = DocSign.GetDocSign(0, "123456", 2)

Отримання підписантів документа

GetDocSignInfo(CardCode As Long, ExDocID As String) As IZDataset

Вхідні параметри:

CardCode – внутрішній ідентифікатор документа у М.Е.Doc; **ExDocID** – зовнішній ідентифікатор документа (необов'язковий).

Відповідь:

IZDataset - таблиця з інформацією про підписантів документа:

IPN As String – IПН; FIO As String – кому видано сертифікат; ISSUER As String – Видавець сертифікату; SERIAL As String – ідентифікатор ключа; DATEBEG As Date – дата початку дії сертифікату; DATEEND As Date – дата закінчення сертифікату; TITLE As String – посада; SN As String – серійний номер сертифікату; SIGNLEVEL As Long – тип сертифікату: «-1» - «не визначено»

- «0» «директор»
- «1» «бухгалтер»
- «2» «печатка»
- «3» «співробітник»
- «5» «адміністратор реєстрації»
- «7» «фізична особа»

Приклад:

```
set App = CreateObject("Medoc.ZApplication")
set DocSign = App.GetDocSigning()
Signs = DocSign.GetDocSignInfo( 123456 )
'aбo
Signs = DocSign.GetDocSignInfo( 0, "123456" )
```

signlevel = Signs.Fields.Item("SIGNLEVEL").Value
fio = Signs.Fields.Item("FIO").Value

Отримання списку операцій та квитанцій по документу

GetDocKVT(ExDocID As String, CardCode As Long) As IZDataset

Вхідні параметри:

ExDocID – зовнішній ідентифікатор документа; **CardCode** – внутрішній ідентифікатор документа у БД М.Е.Doc.

Відповідь:

Об'єкт таблиця <u>IZDataset</u> з полями: **OPERDATE** - дата операції; **OPERTYPE** - тип операції з Перелік <u>OperType</u>; **KVT_TEXT** - текст квитанції (не підписаної); **KVT_SIGN** - текст квитанції з підписами; REGDATE - дата реєстрації; REGNUM - реєстраційний номер; REASON - причина відхилення; FILENAME – ім'я файла;

Приклад:

Loop

Перелік OperType As Long:

- 0 Готується;
- 1 Підписано бухгалтером;
- 2 Підписано директором;
- 3 Підписано печаткою;
- 4 Відправлено;
- 5 Збережено на диск;
- 6 Отримано повідомлення;
- 7 Отримана перша квитанція;
- 8 Отримана друга квитанція;
- 9 Здано;
- 10 Не прийнято;
- 11 Помилка відправки;
- 12 На підписі;
- 13 Архів;
- 14 Отримано;
- 15 Імпорт;
- 16 Відхилено;
- 17 Підписано співробітником;
- 18 Обговорити;
- 19 Доставлено;
- 20 Прийнятий у роботу;
- 21 Повернуто для змін;
- 22 Відправлено на мобільний телефон;
- 23 Підписано адміністратором регістрації;
- 24 Містить всі підписи;

- 25 Погоджується (не використовується);
- 26 Договір виконано;
- 27 Договір анульовано;
- 28 Договір розірвано;
- 35 Отримана третя квитанція;
- 44 Відправка дубліката;
- 47 Підписано відп. особою;
- 48 Очікування квитанції №2;
- 49 Відповідь підписано бухгалтером;
- 50 Відповідь підписано директором;
- 51 Відповідь підписано співробітником;
- 52 Відповідь підписано печаткою;
- 53 Заблоковано відправником;
- 55 Повідомлення;
- 56 Квитанція №1;
- 57 Зареєстровано в РТТН;
- 58 Повний обсяг;
- 59 Фактичний обсяг;
- 60 Невдале блокування;
- 61 Вдале блокування;
- 62 Відправлено посилання;
- 110 Реєстрація зупинена;
- 111 Рішення СМКОР про реєстрацію;
- 112 Рішення СМКОР про відмову;
- 113 Рішення про розгляд скарги.

Отримання протоколу руху документа

GetDocProtocol(ExDocID As String, CardCode As Long) As String

Метод дозволяє за зовнішнім ідентифікатором **ExDocID** або за внутрішнім **CardCode** отримати протокол руху документа.

Файл протоколу по документу (у форматі .pdf) повертається у вигляді строки в base64.

Вхідні параметри:

ExDocID – зовнішній ідентифікатор документа; **CardCode** – внутрішній ідентифікатор документа у М.Е.Doc (опційний);

Відповідь:

Строка - протоколу руху документа в base64.

Приклад:

set App = CreateObject("Medoc.ZApplication")
set prim = App.GetPrimaryDocs()

retProtoByExdocid = prim.GetDocProtocol("1234")

Отримання статусу руху документа по ExDocID

GetSendSTTByExDocID(ExDocID As String) As Long

Вхідні параметри:

ExDocID – зовнішній ідентифікатор документа

Відповідь:

Число - статус із Перелік <u>OperType</u> або -1 якщо документ не знайдено

Приклад використання в 1С:

//рекомендується перевіряти документ на наявність перед його перезаписом. Наприклад: Если App.GetSendSTTByExDocID(ExDocID)= -1 Тогда

Сообщить("Документ не передавався у М.Е.Doc!");

//створюємо документ

Иначе

Сообщить("Документ знайдено М.Е.Doc!");

//в залежності від необхідності, можемо зробити Возврат або переміщення у Корзину існуючого документа методом <u>MoveToTrash()</u>, наприклад задавши Вопрос() користувачу або визначивши поведінку програмно, чи задавши налаштування користувацького інтерфейсу. Якщо документ переміщено в Корзину – то він позбавляється ExDocID. Таким чином, у системі буде завжди тільки один документ з унікальним ІД (в межах організації).

КонецЕсли;

Отримання статусу руху документа по CardCode

GetSendSTTByCode(CardCode As Long) As Long

Вхідні параметри:

CardCode – внутрішній ідентифікатор документа у БД М.Е.Doc.

Відповідь:

Число - статус із Перелік <u>OperType</u> або -1 якщо документ не знайдено

Приклад:

set App = CreateObject("Medoc.ZApplication")
DocStt = App.GetSendSTTByCode(1234)

Переміщення документа у корзину по ExDocID MoveToTrash(ExDocID As String)

Вхідні параметри:

ExDocID – зовнішній ідентифікатор документа

Приклад:

set App = CreateObject("Medoc.ZApplication")
App.MoveToTrash("1234")

Переміщення документа у корзину по CardCode MoveToTrashByCode(CardCode As Long)

Вхідні параметри:

CardCode – внутрішній ідентифікатор документа у БД М.Е.Doc.

Приклад:

```
set App = CreateObject("Medoc.ZApplication")
App.MoveToTrashByCode(1234)
```

Перевірка наявності документа у корзині по ExDocID

GetInTrashStatus(ExDocID As String) As Boolean

Вхідні параметри:

ExDocID – зовнішній ідентифікатор документа

Відповідь:

True – в корзині; False - не в корзині.

Приклад використання в 1С:

Если App.GetSendSTTByExDocID(IDDoc)<> -1 Тогда

Если App.GetInTrashStatus(IDDoc) = 0 Тогда

Сообщить ("Документ не у Корзині, виконуємо переміщення");

App.MoveToTrash(IDDoc);

Сообщить ("Перемістили у Корзину");

Иначе

Сообщить ("Документ вже у Корзині");

КонецЕсли;

Иначе

Сообщить ("Документа нема у програмі"); КонецЕсли:

Перевірка наявності документа у корзині по CardCode GetInTrashStatusByCode(CardCode As Long) As Boolean

Вхідні параметри:

CardCode – внутрішній ідентифікатор документа у БД М.Е.Doc.

Відповідь:

True – в корзині; False - не в корзині.

Приклад:

set App = CreateObject("Medoc.ZApplication")
IsInTrash = App.GetInTrashStatusByCode(1234)

Отримання статусу документа по ExDocID

GetDocStatus(ExDocID As String) As Long

Вхідні параметри:

ExDocID – зовнішній ідентифікатор документа

Відповідь:

Число - статус із Перелік <u>CARD.STATUS</u>

Приклад:

set App = CreateObject("Medoc.ZApplication") DocStatus = App.GetDocStatus("1234")

Перелік статусу документа CARD.STATUS As Long:

- «0» документ не знайдено;
- «1» чернетка;
- «2» перевірений;
- «З» помилковий;
- «4» імпортований;
- «5» копія;
- «6» надісланий;
- «7» контроль відключено.

Отримання статусу документа по CardCode

GetDocStatusByCode(CardCode As Long) As Long

Вхідні параметри:

CardCode – внутрішній ідентифікатор документа у БД М.Е.Doc.

Відповідь:

Число - статус із Перелік <u>CARD.STATUS</u>

Приклад:

set App = CreateObject("Medoc.ZApplication")

Отримати дату створення документа в М.Е.Doc

GetDocCrtDate(ExDocID As String) As Date

Вхідні параметри:

ExDocID – зовнішній ідентифікатор документа

Відповідь:

Дата - дата створення документа у БД М.Е.Doc

Приклад:

set App = CreateObject("Medoc.ZApplication")
DocCrtDate = App.GetDocCrtDate("1234")

Отримати реєстраційний номер документа ПН/РК у ДПС GetNaklRegNum(ExDocID As String) As String

Вхідні параметри:

ExDocID – зовнішній ідентифікатор документа

Відповідь:

Строка - реєстраційний номер документа у ДПС

Приклад:

set App = CreateObject("Medoc.ZApplication")
set Prim = App.GetPrimaryDocs()
RegNum = Prim.GetNakIRegNum("1234")

Отримати дату реєстрації документа ПН/РК у ДПС

GetNaklRegDate(ExDocID As String) As Date

Вхідні параметри:

ExDocID – зовнішній ідентифікатор документа

Відповідь:

дата - дата реєстрації документа у ДПС

Приклад:

```
set App = CreateObject("Medoc.ZApplication")
set Prim = App.GetPrimaryDocs()
RegDate = Prim.GetNakIRegDate("1234")
```

Отримання Витягу з ЄРПН F/J1400107

GetNNResponse(exDocIDRequest As String, docNumRequest As String, exDocIDNN As String) As IZDocument

Вхідні параметри:

Метод викликається з вказанням тільки одного із трьох параметрів, інші залишаються порожніми: exDocIDRequest – зовнішній ідентифікатор запиту; docNumRequest – номер запиту; exDocIDNN – зовнішній ідентифікатор документа, на який відправлено запит.

Відповідь:

Об'єкт документа витягу IZDocument

Приклад:

```
set App = CreateObject("Medoc.ZApplication")
```

```
set GovReq = App.GetGovRequest()
set DocResp = GovReq.GetNNresponse("", "1234", "")
'відкриємо таблицю отриманих із витягу
```

```
set DsTab1 = DocResp.DataSets("TAB1")
DsTab1.First()
do While Not DsTab1.Eof
regDate = DsTab1.Fields.Item("TAB1_A2").Value
regNum = DsTab1.Fields.Item("TAB1_A9").Value
docName = DsTab1.Fields.Item("TAB1_A5").Value
docNum = DsTab1.Fields.Item("TAB1_A31").Value
docDate = DsTab1.Fields.Item("TAB1_A4").Value
'...
DsTab1.Next()
```

Loop

Отримати останній Витяг з ЄРПН за фільтром даних документа у Витягу GetNNResponseNN(nnDS As IZDataset) As IZDocument

Вхідні параметри:

 IZDataset
 – таблиця параметрів документа з полями:

 DOCTYPE*
 – тип документа : ПН = 1; РК = 2;

 DOCDATE*
 – дата документа;

 NN_NUM1*
 – номер документа;

 NN_NUM2
 – номер документа (вид діяльності);

 NN_NUM3
 – номер документа (номер філії);

 IPN
 – ППН продавця.

Відповідь:

set App = CreateObject("Medoc.ZApplication") створюємо порожній датасет з полями пошуку set nnDS = App.GetEmptyDataSet() nnDS.Fields.AddField "DOCTYPE", 0 nnDS.Fields.AddField "DOCDATE", 4 nnDS.Fields.AddField "NN NUM1", 2 nnDS.Fields.AddField "IPN", 2 заповнюємо рядок датасету nnDS.Edit() nnDS.Fields.Item("DOCTYPE") = 1 nnDS.Fields.Item("DOCDATE") = CDate("27.11.2020") nnDS.Fields.Item("NN_NUM1") = "1" nnDS.Fields.Item("IPN") = "112233445566" пошук витягу за параметрами документа set DocResp = App.GetGovRequest().GetNNResponseNN(nnDS) виведемо наприклад номер витягу set main = DocResp.DataSets(0)

GovRespNum = main.Fields.Item("N1").Value

Отримання списку Витягів F/J1400107 в заданому діапазоні

GetNNResponseList(datebeg As Date, dateend As Date, num_beg As Long, num_end As Long) As IZDataset Вхідні параметри:

datebeg – початкова дата діапазону для вибору витягів;

dateend - кінцева дата діапазону для вибору витягів;

num_beg - початковий номер витягу;

num_end - кінцевий номер витягу;

Якщо num_beg або num_end вказати 0, то пошук за цими параметрами не виконуватиметься.

Відповідь:

IZDataset – таблиця витягів з полями:

CardCode As Long – внутрішній ідентифікатор документа у М.Е.Doc; RepNum As Long – номер витягу; RepDate As Date – дата витягу;

Приклад:

set App = CreateObject("Medoc.ZApplication")
set GovReq = App.GetGovRequest()
set ResponseList = GovReq.GetNNresponseList("01.10.2020", "01.11.2020", 0, 0)

ResponseList.First() 'перший запис таблиці cardcode = ResponseList.Fields.Item("CardCode").Value repnum = ResponseList.Fields.Item("RepNum").Value repdate = ResponseList.Fields.Item("RepDate").Value

(Відкрити потрібний витяг можна методом **OpenDocumentByCode**(**CardCode** As Long) – відкриття документа за ідентифікатором)

Отримання статусу реєстрації ПН/РК в ДПС

FillNNStatusDS(filterDs As IZDataset) As IZDataset

Метод доповнює передану таблицю filterDs інформацією про реєстрацію документів на основі заповнених полів про дані документа в filterDs.

Вхідні параметри:

IZDataset – таблиця документів з полями: **DOC_ID** – зовнішній ідентифікатор документа (ExDocID); **DOC ТҮРЕ*** – тип документа : 1 - ПН, 2 - РК до ПН; **DOC_MOVE*** – напрямок : 1 - видані, 2 - отримані; **DOC_NUM** – номер документа; **DOC VD** - код діяльності; **DOC_DEPT** - номер філії; **DOC_DATE** – дата документа; **FIRM_EDRPOU** – ЄДРПОУ підприємства; **PARTNER_EDRPOU** – ЄДРПОУ покупця; **PATRNER_IPN** – IПН покупця; **VATSUM** - сума ПДВ; **REGDATE** – дата реєстрації; **STATUS** – статус реєстрації документа в ДПС, можливі значення: «-З» - відсутній обов'язковий параметр; «-2» - критеріям пошуку відповідає більше одного документа; «-1» - відсутній документ; «0» - не відправлявся на реєстрацію; «1» - відправлено в ДПС; «2» - доставлено в ДПС; «З» - не прийнято в ДПС; «4» - зареєстровано в ДПС.

DOC_TYPE та **DOC_MOVE** - обовязкові параметри до заповнення;

Відповідь:

IZDataset – дозаповнена вхідна таблиця документів з відповідними полями.

Приклад:

set App = CreateObject("Medoc.ZApplication") set filterDS = App.GetEmptyDataSet() 'створення порожньої таблиці filterDS.Fields.AddField "DOC ID", 2 filterDS.Fields.AddField "DOC TYPE", 2 filterDS.Fields.AddField "DOC_MOVE", 2 filterDS.Fields.AddField "DOC NUM", 2 filterDS.Fields.AddField "DOC_VD", 2 filterDS.Fields.AddField "DOC_DEPT", 2 filterDS.Fields.AddField "DOC DATE", 2 filterDS.Fields.AddField "FIRM_EDRPOU", 2 filterDS.Fields.AddField "PARTNER_EDRPOU", 2 filterDS.Fields.AddField "PATRNER IPN", 2 filterDS.Fields.AddField "OSOBFIO", 2 filterDS.Fields.AddField "VATSUM", 2 filterDS.Fields.AddField "REGDATE", 2 filterDS.Fields.AddField "STATUS", 2 filterDS.First() filterDS.Edit() filterDS.Fields.Item("DOC_ID").Value = "1234" filterDS.Fields.Item("DOC_TYPE").Value = "1" '- обяв'язковий до заповнення filterDS.Fields.Item("DOC_MOVE").Value = "1" '- обяв'язковий до заповнення set resDS = App.FillNNStatusDS(filterDS)

resDS.First()

```
VATSUM= resDS.Fields.Item("VATSUM").Value
REGDATE = resDS.Fields.Item("REGDATE").Value
STATUS = resDS.Fields.Item("STATUS").Value
```

2.3. Відправка, погодження/відхилення документів

Помістити в список до відправки документ по ExDocID Add(ExDocID As String)

```
Вхідні параметри:
```

ExDocID – зовнішній ідентифікатор документа.

Приклад:

set Sender = GetDocSender() Sender.Add("1234")

Помістити в список до відправки документ по DocID AddDocID(DocID As String)

Вхідні параметри:

DocID – унікальний внутрішній ідентифікатор документа в рамках документообміну.

Приклад:

set Sender = GetDocSender()
Sender.AddDocID("1234")

Помістити в список для погодження/відхилення документа

AcceptDoc(DocID As String, Accept As Boolean, Reason As String)

Для вказаного документа виконається формування повідомлення про погодження/відхилення документа

Вхідні параметри:

DocID – унікальний внутрішній ідентифікатор документа в рамках документообміну; **Accept** – ознака погодження/відхилення документа - true/false (1/0); **Reason** – текст причини відхилення документа.

Приклад:

```
set Sender = GetDocSender()
Sender.AcceptDoc("1234", 1, "")
'або відхилення документа
Sender.AcceptDoc("1234", 0, "Невідповідність найменування товару")
```

Очистити список документів до відправки Clear()

Приклад:

```
set Sender = GetDocSender()
Sender.Clear()
```

Виконати відправку документів/повідомлень зі сформованого списку

Send(toGovAdr As Boolean) As Long

Зауваження. Реалізована підтримка відправки первинних документів в форматі *.rtf, податкових накладних і додатку 2 ПН, акцизних накладних і додатку 2 АН, а також відправка інформаційних довідок.

Вхідні параметри:

toGovAdr – ознака відправки документів/повідомлень в контролюючий орган/контрагенту - true/false (1/0).

Якщо можливий тільки один отримувач, програма відправляє документ автоматично, інакше - в інтерфейсі з'явиться вікно вибору отримувача.

Відповідь:

Число - кількість відправлених документів.

Приклад:

```
set App = CreateObject("Medoc.ZApplication")
set Sender = GetDocSender()
Sender.Add("1234")
```

'попередньо необхідно помістити кеш програми комплект підписів документів та шифрування методом <u>CertToCache()</u> для виконання операції без взаємодії з інтерфейсом М.Е.Doc для вводу паролей підписів

SndCount = Sender.Send(0)

Довідники

Приклади довідників які доступні через СОМ : **HBPARTNER** – контрагенти; **HBPARTNERGRP** – групи контрагентів; **HBGOODS** – номенклатура товарів та послуг; **HBGOODSGRP** – групи номенклатур товарів та послуг; **PHYSPERSONS** – співробітники; **HBTAXINSP-** податкові інспекції; **HBVO** – відповідальні особи.

Деякі довідники ведуться в розрізі груп (контрагенти, номенклатура товарів та послуг), тому попередньо необхідно в довіднику груп - створити групу, та при записі в довідник вказувати ідентифікатор групи для запису, інакше вони не будуть відображені у програмі.

Деякі довідники ведуться у розрізі організацій : групи контрагентів, групи номенклатур товарів та послуг, співробітники, відповідальні особи. Тому при роботі з такими довідниками рекомендується додати умову перевірки по полю IDORG, значення якого можна отримати функцією : <u>SeekOrg()</u>

Для роботи з Довідниками необхідно використовувати функцію GetDictionary()

Отримання інформації з Довідника

Open(DictName As String) As IZDataset

Якщо довідник ведеться в розрізі організації - то буде повернуто довідник відкритої організації в M.E.Doc

Вхідні параметри:

DictName – назва довідника;

Відповідь:

IZDataset – таблиця з полями довідника.

Приклад:

```
set App = CreateObject("Medoc.ZApplication")
set Dict = App.GetDictionary()
set tabDS = Dict.Open("HBGOODS")
```

Перевірка доступу до довідника тільки на читання(без редагування)

IsReadOnly() As Boolean

Відповідь:

True – довідник тільки на читання; False - довідник доступний на читання та редагування.

Приклад:

```
set App = CreateObject("Medoc.ZApplication")
set Dict = App.GetDictionary()
set tabDS = Dict.Open("HBGOODS")
canread = Dict.IsReadOnly()
if canread then
WScript.Echo "доступ тільки на читання"
if not canread then
WScript.Echo "доступ на читання та редагування"
```

Отримати значення генератора внутрішнього ID запису в довіднику

GetGenID(DictName As String) As Long

Використовується для додавання послідовної коректної нумерації внутрішнього ІД записів без їх дублювання.

При додаванні записів у Довідник - використання обов'язкове! Зазвичай ІД - поле CODE.

Вхідні параметри:

DictName – назва довідника.

Відповідь:

Число – новий ідентифікатор запису в БД, більший на 1 за попередньо повернуте значення.

Приклад:

set App = CreateObject("Medoc.ZApplication")
set Dict = App.GetDictionary()
newid = Dict.GetGenID("HBGOODS")

Видалити всі записи в довіднику Clear(DictName As String)

Використовується для видалення всіх записів із довідника. Якщо довідник ведеться в розрізі організацій - то видаляються записи відкритої організації в М.Е.Doc, загальні довідники очищаються повністю.

Вхідні параметри:

DictName – назва довідника.

Відповідь:

Число – новий ідентифікатор запису в БД, більший на 1 за попередньо повернуте значення.

Приклад:

```
set App = CreateObject("Medoc.ZApplication")
set Dict = App.GetDictionary()
Dict.Clear("HBGOODS")
```

Зберегти зміни в довіднику

Save() As Boolean

Відповідь:

True – зміни в довіднику збережено; False - зміни в довіднику не збережено.

Приклад:

```
set App = CreateObject("Medoc.ZApplication")
set Dict = App.GetDictionary()
set grp = Dict.Open("HBGOODSGRP")
```

```
grp.Edit()
grp.Append() '- додавання запису
```

```
newid = Dict.GetGenId("HBGOODSGRP")
orgId = App.SeekOrg("11121314", "")
```

```
grp.Fields.Item("CODE").Value = newid
grp.Fields.Item("NUM").Value = "1"
grp.Fields.Item("NAME").Value = "Товари"
grp.Fields.Item("IDORG").Value = orgId
```

```
saved = Dict.Save()
if saved then
WScript.Echo "збережено"
if not saved then
WScript.Echo "не збережено"
```

2.4. Шаблони Первинних документів

Отримання списку шаблонів первинних документів

GetTemplates(partid As Long, groupid As Long) As IZDataset

Вхідні параметри:

partId – параметр зарезервовано - приймає значення 7; groupId – параметр зарезервовано - приймає значення 0;

Відповідь:

IZDataset – таблиця зі списом шаблонів та їх властивостей.

Приклад використання в 1С:

```
IPrim = App.GetPrimaryDocs();
templates = IPrim.GetTemplates(7, 0);
Пока HE templates.EOF Цикл
curfid = 0;
Пока curfid < templates.Fields.Count Цикл
Cooбщить( templates.Fields.Item( curfid ).Name + " : " + templates.Fields.Item( curfid
).Value);
curfid = curfid + 1;
КонецЦикла;
Cooбщить( "------");
templates.Next();
КонецЦикла;
```

Отримання XML опису полів шаблона

GetTemplate(partId As Long, templateId As String) As String

Вхідні параметри:

partId – параметр зарезервовано - приймає значення 7; templateld – чаркод шаблона;

Відповідь:

Строка – XML опис полів шаблона та їх властивостей. У тегах **partshema** описуються динамічні поля таблиць.

Приклад використання в 1С:

```
template = IPrim.GetTemplate( 7,"1C82ABP" ); //Акт наданих послуг. Вивід тексту XML
Чтение = Новый ЧтениеXML;
Чтение.УстановитьСтроку( template );
Пока Чтение.Прочитать() Цикл
```

// Прочитати "структурні частини" елементів. Перевірити яка частина елементу - поточна. Если Чтение. ТипУзла = ТипУзлаХМL. НачалоЭлемента И Чтение. Имя = "field" Тогда ИмяУзла = Чтение.Имя; Сообщить("--" + ИмяУзла); // Атрибути елементів можна читати тільки, якщо поточна частина - початок елемента Пока Чтение. Прочитать Атрибут() Цикл // Прочитати дані вузла атрибуту. ТипУзла = Чтение.ТипУзла; Имя = Чтение.Имя; Значение = Чтение.Значение; Сообщить(Имя + ": " + Значение); КонецЦикла; ИначеЕсли Чтение.ТипУзла = ТипУзлаХМL.Текст Тогда // У прикладі просто виводимо текст у вікно повідомлення. Сообщить("Текст : " + Чтение.Значение); ИначеЕсли Чтение.ТипУзла = ТипУзлаХМL.КонецЭлемента Тогда // У прикладі просто виводимо ознаку кінця елементу Сообщить("--Кінець : " + Чтение.Имя); КонецЕсли: КонецЦикла; //-----приклад отримання полів динамічної таблиці-----У отриманій XML схемі у тегах partshema описані динамічні поля. <partshema part="TAB1" caption="Tаблиця 1"> <field name="TAB1 FIELD1" type="String" caption="Поле 1 таб.TAB1" maxlength="0" spr="" sprname="" /> <field name="TAB1 FIELD2" type="String" caption="Поле 2 таб. TAB1" maxlength="0" spr="" sprname="" /> </partshema>

2.5. Додаткові методи

Отримати версію М.Е.Doc

GetPrgVersion() As Long

Відповідь:

число – версія М.Е.Doc без крапок.

В інтерфейсі програми версія має текстовий формат з роздільником – крапка. Тому для порівняння необхідно видалити крапки або виділити декілька символів (пропонується останні 3).

Приклад використання в 1С:

Попытка

ВерсияМедка=App.GetPrgVersion(); //1101001 ВерсияМедкаЧитабельная="11.01." + Прав(ВерсияМедка, 3); Если Число(Прав(ВерсияМедка, 3)) < 024 Тогда Сообщить ("Для коректної роботи необхідна версія не нижче 11.01.024" + Символы.ПС + "У Вас встановлена версія " + ВерсияМедкаЧитабельная + Символы.ПС + "Оновіть програму!");

КонецЕсли;

Возврат;

Исключение

Сообщить("Помилка при визначенні версії"); КонецПопытки:

Отримати ідентифікатор організації в М.Е.Doc

SeekOrg(EDRPOU As String, DEPT As String) As Long

Вхідні параметри:

EDRPOU – ЄДРПОУ організації; **DEPT –** код філії організації.

Відповідь:

Число – внутрішній ідентифікатор організації в М.Е.Doc. Якщо організації не існує, повертається 0.

Приклад використання в 1С:

ИдентификаторПредприятие = App.SeekOrg(ЕДРПОУ, Филиал);

Перехід у вибрану організацію

SelectFirm(EDRPOU As String, DEPT As String) As Boolean

Важливо знаходитися у потрібній організації при використанні функцій створення документів, які НЕ містять вхідного параметра вказуючого на внутрішній ідентифікатор організації, який отримується методом : <u>SeekOrg()</u>.

При створенні документа, наприклад, методами <u>OpenOrCreateByExDocID()</u>, <u>CreateNewRnn()</u> - документи будуть створені у організації яка відкрита в М.Е.Doc.

Якщо у М.Е.Doc не передбачається введення більше 1 підприємства, цей метод можна не використовувати.

Вхідні параметри:

EDRPOU – ЄДРПОУ організації; DEPT – код філії організації.

Відповідь:

True – перехід виконано; *False* – перехід не виконано.

Приклад використання в 1С:

ОпределимПредприятие = App.SelectFirm(ЕДРПОУ, Филиал); Если ОпределимПредприятие Тогда Doc = App.OpenOrCreateByExDocID(Бланк, 0, PDate, , ИД, 0);

Отримати ЄДРПОУ та код філіалу відкритої організації в М.Е.Doc

GetCurrEdrpou() As String

Відповідь:

Строка - ЄДРПОУ відкритої організації в М.Е.Doc

GetCurrDept() As String

Відповідь:

Строка - код філіалу відкритої організації в М.Е.Doc

Рекомендується дані методи використовувати комплексно - для визначення відкритої організації, наприклад при створенні документа.

Приклад використання в 1С:

App = Новий СОМОб'єкт("MEDOC.ZApplication");

CurrEdrpou = App.GetCurrEdrpou();

CurrDept = App.GetCurrDept();

CurrOrg = CurCurrEdrpou + CurrDept;

Организацияв1С = КодЕДРПОУ + ЗначениеФилиала;

Попытка

Если CurrOrg <> Организацияв1С Тогда

Сообщить ("Документи не можуть бути передані! Вибрана організація у програмі : " + Организацияв1С + ", а у М.Е.Doc відкрито : " + CurrOrg); //якщо не співпадають, не передаємо документи

//якщо не співпадають, не передаємо документі

Возврат;

КонецЕсли;

Исключение

Сообщить("Помилка при визначенні організації");

КонецПопытки;

Отримати список документів за умовою

DocumentsDataSet(CondString As String) As IZDataset

Вхідні параметри:

CondString – рядок умови пошуку з використанням стандартного синтаксису SQL оператора WHERE.

Параметри, доступні для використання в запитах:

IDREPCODE - код відповіді в

IDPARENT - ознака батьківського документу. Приймає значення 0, 1.

FORM - код форми документу в БД Медку. Наприклад, 21340 відповідає податковій накладній J1201013.

СНАRCODE - чаркод форми документу. Наприклад J1201013 - податкова накладна.

CRTDATE - дата створення документу або дата з точністю до секунд. Наприклад, «2023-02-01», «2023-02-01 17:32:13».

MODDATE - дата останньої модифікації документу. Початкове значення дорівнює даті створення. PERDATE - дата документу.

STATUS - статус документу з переліку:

- 1 Новий
- 2 Перевірений
- 3 Помилковий
- 4 Імпортований
- 5 Копія

FLAGS - внутрішня ознака документу.

SENDSTT - статус із Переліку OperType або 1.

Приклади можливих запитів:

```
query = "IDREPCODE = '1''"
query = "IDPARENT >0"
query = "FORM = '21340'"
query = "CHARCODE = 'J1201013'"
query = "CRTDATE > '01.02.2023' AND CRTDATE < '01.03.2023 15:00:00' "
query = "MODDATE > '01.02.2023' AND MODDATE < '01.03.2023' "
query = "PERDATE = '01.02.2023'"
query = "STATUS = '2'"
query = "FLAGS = '0''
query = "SENDSTT = '9'"
set Documents = App.DocumentsDataSet(query)</pre>
```

Відповідь:

IZDataset – таблиця документів згідно умов пошуку.

Приклад:

set App = CreateObject("Medoc.ZApplication")
set DS = App.DocumentsDataSet("CRTDATE > '01.02.2023' and CHARCODE = 'J1201010'')

Отримати повідомлення

ReceiveCorrespondence() As Boolean

Метод виконує отримання повідомлень, документів та квитанцій з контролюючих органів, СДО та налаштованої електронної пошти (аналог F9 у інтерфейсі)

Відповідь:

True – отримано повідомлення; False – нові повідомлення відсутні.

Приклад:

set App = CreateObject("Medoc.ZApplication") '- попередньо рекомендується додати сертифікат шифрування відповідної організації в кеш програми методом <u>CertToCache()</u> - для виконання отримання повідомлень без вводу пароля в інтерфейсі IsNewMsg = App.ReceiveCorrespondence()

Запис у кеш програми сертифіката, секретного ключа та пароля

CertToCache(certPath As String, skPath As String, skPwd As String) As Long

Вхідні параметри:

certPath – повний шлях до файлу сертифіката; skPath – повний шлях до файлу секретного ключа; skPwd – пароль до пари certPath – skPath; *у випадку використання <u>захищених носіїв</u> - параметр skPath вказується порожнім.

Відповідь:

число – код статусу виконання:

- «0» вдалось успішно записати в кеш;
- «1» за вказаним шляхом не знайдено сертифікат;
- «2» за вказаним шляхом не знайдено секретний ключ;
- «3» вказано неправильний пароль;
- «4» не співпадає пара сертифікат секретний ключ;
- «5» вказаний сертифікат відсутній у М.Е.Doc;
- «6» вказаний сертифікат не належить організації яка відкрита у М.Е.Doc;
- «7» відсутній підключений/потрібний захищений носій;
- «8» неправильний пароль, залишилась остання спроба;
- «9» неправильний пароль, захищений носій заблокований.

Приклад:

set App = CreateObject("Medoc.ZApplication") set Sign = App.GetDocSigning() resultCode = sign.CertToCache("C:\Keys\cert.crt", "C:\Keys\key.zs2", "111") if(resultCode != 0) Then 'обробка помилки додавання в кеш

Перевірка платників ПДВ SentGetNDSPLTSTT(filterDs As IZDataset) As IZDataset

Метод доповнює передану таблицю filterDs інформацією про статус платника ПДВ «STATUS» та результату перевірки «RESULT».

Якщо по данному ІПН відсутня інформація в М.Е.Doc, то виконається відправка запиту в ДПС по заданим даним.

Вхідні параметри:

IZDataset – таблиця документів із полями:

IPN* – IПН контрагента який перевірятиметься;

PERDATE* – дата, станом на яку необхідна перевірка статусу платника ПДВ;

STATUS – можливі значення :

- «0» відправлено запит в ДПС;
- «1» платник ПДВ;
- «2» неплатник ПДВ.

RESULT – можливі значення :

- «0» ІПН контрагента відсутній в реєстрі (некоректний);
- «1» ІПН входить в список на вказану дату перевірки;
- «2» ІПН не входить в список на вказану дату перевірки;
- «3» інформація по ІПН відсутня в базі.

Відповідь:

IZDataset – дозаповнена вхідна таблиця документів в полях STATUS та RESULT.

Приклад:

set App = CreateObject("Medoc.ZApplication") set filterDS = App.GetEmptyDataSet() 'створення порожньої таблиці

filterDS.Fields.AddField "IPN", 2 filterDS.Fields.AddField "PERDATE", 4 filterDS.Fields.AddField "STATUS", 2 filterDS.Fields.AddField "RESULT", 2

filterDS.First() filterDS.Edit() filterDS.Fields.Item("IPN").Value = "123456789012" '- обяв'язковий до заповнення filterDS.Fields.Item("PERDATE").Value = "11.11.2020" '- обяв'язковий до заповнення

set primDocs = App.GetPrimaryDocs()
set resDS = primDocs.SentGetNDSPLTSTT(filterDS)

STATUS = resDS.Fields.Item("STATUS").Value RESULT = resDS.Fields.Item("RESULT").Value

Отримання інформації про статус контрагента в реєстрі платників ПДВ

Для отримання інформації про контрагента використовується послідовність методів.

1. Методом **OpenOrCreateByExDocID**() створюється запит до ДПС J1300406 Ф.1-ЗДР Запит про отримання довідки з реєстру платників ПДВ, в якому вказуються контрагенти, по яким необхідно оновити інформацію і заповнюється таблиця з вказанням ІПН контрагентів.

2. Створений запит додається в чергу відправки методом AddDocID() і відправляється в методом Send().

3. Через деякий час необхідно прийняти пошту методом ReceiveCorrespondence().

4. За допомогою метода GetNNResponse() та відомого зовнішнього ідентифікатора запиту EXDOCID серед отриманих документів можна визначити потрібну відповідь на запит - J1400406 Ф.2-ДР Довідка з Реєстру платників ПДВ, з подальшим аналізом її вмісту.

Отримати посилання на документ

GetDocWebAdr(ExDocID As String, CardCode As Long) As String

Метод повертає посилання на відправлені первинні документи, що зберігаються на платформі ПТАХ.

Вхідні параметри:

ExDocID – зовнішній ідентифікатор документа; CardCode – внутрішній ідентифікатор документа у БД М.Е.Doc.

Відповідь:

Рядок - посилання на відправлений документ Якщо виникає помилка - повертається пустий рядок.

Приклад:

```
set App = CreateObject("Medoc.ZApplication")
set prim = App.GetPrimaryDocs()
ret = prim.GetDocWebAdr("C533DC9B-06FA-4B48-BE6B-AA76ADD61567", 0) '3a ExDocID
ret = prim.GetDocWebAdr("", 650652) '3a cardCode
```

2.6. Взаємодія з інтерфейсом М.Е.Doc

Відкрити документ в інтерфейсі M.E.Doc за ExDocID

ShowDocument(ExDocID As String)

Якщо відкривається кілька документів у циклі, то у програмі М.Е.Doc вони будуть розміщені зліва направо у порядку виклику метода.

Вхідні параметри:

ExDocID – зовнішній ідентифікатор документа;

Приклад:

set App = CreateObject("Medoc.ZApplication")
App.ShowDocument("123")

Відкрити документ в інтерфейсі M.E.Doc за CardCode ShowDocumentByCode(CardCode As Long)

Якщо відкривається кілька документів у циклі, то у програмі М.Е.Doc вони будуть розміщені зліва направо у порядку виклику метода.

Вхідні параметри:

CardCode – внутрішній ідентифікатор документа у БД М.Е.Doc;

Приклад:

```
set App = CreateObject("Medoc.ZApplication")
```

```
App.ShowDocumentByCode(123)
```

Запуск модуля М.Е.Doc

RunModule(SubSystem As String) As ISubSysFilter

Якщо модуль вже відкритий – переміщення фокусу на вкладку. Модулі SubSystem: DocRst - Звітність - Реєстр звітів FicsalAccTaxInvoices - Облік ПДВ - Реєстр виданих та отриманих податкових накладних TaxDocsRstNew - Первинні документи - Реєстр податкових документів PrimaryDocsRstNew - Первинні документи - Реєстр електронних документів" GovQry - Довідки державних органів - Інформаційна довідка AdmCert - Адміністрування - Сертифікати - Встановлені сертифікати

Відповідь:

ISubSysFilter – інтерфейс роботи з фільтрами відображення документів модулю : <u>GetFilter()</u> As IZDataset - Отримати фільтри відкритого модулю ApplyFilter() - Застосувати фільтр для модулю

Приклад:

set App = CreateObject("Medoc.ZApplication")

set SubSysFilter = App.RunModule("PrimaryDocsRstNew")

Отримати фільтри відкритого модулю

ISubSysFilter.GetFilter() As IZDataset

Отримати таблицю фільтру для підсистеми для заповнення параметрами фільтрації. Якщо для підсистеми не задано фільтр, то повернеться пуста таблиця.

Відповідь:

Об'єкт таблиця IZDataset з полями фільтрів модуля

Приклад використання в 1С:

Приклад використання для Реєстру звітів: Арр = Новый СОМОбъект("MEDOC.ZApplication"); reestrFil = App.RunModule("DocRst"); //запуск модуля та отримання інтерфейсу фільтру filterTbl = reestrFil.GetFilter(); //отримати фільтр Если filterTbl.Fields.Count = 0 Тогда Сообщить("Для цього модулю фільтр не визначений!");

Возврат;

КонецЕсли;

filterTbl.Fields.Item("PERDATE").Value = "01.11.2020"; //за 2018 filterTbl.Fields.Item("PERTYPE").Value = -1; //всі filterTbl.Fields.Item("PART").Value = 0; //Податкова filterTbl.Fields.Item("SENDSTT").Value = "Rejected"; //Не прийняті reestrFil.ApplyFilter();

Приклад використання для Реєстру податкових документів:

App = Новый СОМОбъект("MEDOC.ZApplication");

reestrFil = App.RunModule("PrimaryDocsRstNew");

filterTbl = reestrFil.GetFilter(); //отримати фільтр

Если filterTbl.Fields.Count = 0 Тогда

Сообщить("Для цього модулю фільтр не визначений!");

Возврат;

КонецЕсли;

filterTbl.Fields.Item("DOCTYPE").Value = 10100; //тип документа з переліку <u>docType</u> filterTbl.Fields.Item("PERFROM").Value = "01.11.2020"; //період з filterTbl.Fields.Item("PERTO").Value = "01.12.2020"; //період по filterTbl.Fields.Item("PARTNEREDRPOU").Value = "112233445566"; //ЄДРПОУ контрагента filterTbl.Fields.Item("PARTNERDEPT").Value = ""; //Код філії контрагента filterTbl.Fields.Item("ROUTE").Value = 2; //1 – вхідні, 2 - вихідні filterTbl.Fields.Item("SUBROUTE").Value = -3; //«-3» на опрацюванні, «-2» - архів, «-1» - корзина reestrFil.ApplyFilter();

//Приклад використання для Інформаційної довідки:

App = Новый СОМОбъект("MEDOC.ZApplication"); reestrFil = App.RunModule("GovQry"); filterTbl = reestrFil.GetFilter(); //отримати фільтр Если filterTbl Fields.Count = 0 Тогда

Сообщить("Для цього модулю фільтр не визначений!");

Возврат;

КонецЕсли; filterTbl.Fields.Item("PERDATE").Value = "01.11.2020"; //період reestrFil.ApplyFilter();

Програмно «Оновити» вікно з переліком документів у M.E.Doc RefreshRstDoc()

Приклад використання в 1С:

//передали документ, тепер оновимо інтерфейс М.Е.Doc, для того щоб новий документ відобразився, інакше доведеться натискати однойменну кнопку, щоб документ відобразився App.RefreshRstDoc();

Сообщить("Документ переданий. Перейдіть у М.Е.Doc");

2.7. IZDocument - об'єкт документа

Отримати об'єкт таблиці даних документа DataSets(TblName As Variant, Totals As Long) As IZDataset

Вхідні параметри:

TblName - назва таблиці/індекс "MAIN"/0; "TAB1"/1 - підтаблиця 1 (0 - за замовчуванням); **Totals** - ознака отримання підсумків по вибраній таблиці - 1/0 (0 - за замовчуванням)

Відповідь:

Об'єкт таблиця <mark>IZDataset</mark>

Якщо вказана ознака отримання підсумків - буде повернуто тільки один рядок таблиці

Приклад:

set dsMain = Doc.DataSets("0") ' отримуємо таблицю MAIN set dsTab1 = Doc.DataSets("TAB1", 1) ' отримуємо рядок підсумків по таблиці TAB1

Встановити тип документа

Specif(Specif As Long)

Вхідні параметри:

Specif - встановити для документа необхідний тип:

- 0 Звітний;
- 1 Уточнюючий;
- 2 Звітний новий;
- 3 Довідковий.

Додавати слід перед збереженням Save.

Приклад:

Doc.Specif = 3 'застосувати тип Довідковий для звіту

Doc.Save()

Зберегти документ

Save(Mode As Long)

Вхідні параметри:

Mode - перевіряти документ перед збереженням - 1/0 (Так/Ні). Mode = 1 - аналог натискання "Перевірити документ"/**F4** у документі. (0 - за замовчуванням)

Приклад:

Doc.Save() ' збереження без перевірки

Doc.Save(1) ' збереження з перевіркою

Отримати CardCode ідентифікатор документа в БД М.Е.Doc CardCode As Long

Відповідь:

DocCode = Doc.CardCode

Ідентифікатор головного документу

IDParent As Long

Відображає/встановлює CardCode Ідентифікатор головного документу. При наявності параметру документ вважається дочірнім для документа IDParent. Параметр застосовується тільки для пакетних звітів.

Відповідь:

Число - ідентифікатор головного документу в БД М.Е. Doc

Приклад:

ChildDoc.IDParent = ParentDoc.CardCode

Відключити автоматичний розрахунок документа DisableScripts()

Може бути корисним у випадках, коли програмний розрахунок сум не може бути використаний. Наприклад, сума не повинна розраховуватися як ціна помножена на кількість, так як ціну бухгалтер корегує на копійку (для округлення), і потрібно передавати дані у М.Е.Doc «як є».

Необхідно виконувати перед заповненням документа даними.

Приклад використання в 1С:

Doc = App.OpenOrCreateByExDocID(Бланк, 0, PDate, , IД, 0);

Doc.DisableScripts(); //відключаємо бейсікскрипти

//Створюємо документ

rsMain = Doc.DataSets();

rsMain.Edit();

//заповнення даних

Doc.EnableScripts(); //включаємо скрипти

Doc.Save(1);

Doc.Close();

Включити автоматичний розрахунок документа EnableScripts()

За замовчуванням автоматичний розрахунок увімкнено.

Приклад:

Doc.EnableScripts()

2.8. IZDataSet - об'єкт таблиця

Отримати об'єкт колекції полів поточного рядка таблиці Fields As IZFields

Відповідь:

IZFields - об'єкт колекції полів поточного рядка таблиці

Приклад:

Flds = Ds.Fields

Додати запис в таблицю Append()

Приклад: Ds.Append()

Редагувати таблицю Edit()

Приклад: Ds.Edit()

Номер позиції поточного рядка в таблиці RecNo As Long

Приклад:

position = Ds.RecNo

Кількість записів у таблиці RecordCount As Long

records = Ds.RecordCount

Встановити курсор на перший запис таблиці First()

Зазвичай виконується перед циклом проходу по таблиці.

Приклад:

Ds.First()

Перехід на наступний запис таблиці Next()

Приклад:

Ds.Next()

Перехід на попередній запис таблиці Prior()

Приклад: Ds.Prior()

Перехід на останній запис таблиці Last()

Приклад:

Ds.Last()

Ознака знаходження курсора поза межами початку таблиці Bof As Boolean

Приклад:

isOutsideBegining = Ds.Bof

Ознака знаходження курсора поза межами кінця таблиці Eof As Boolean

isOutsideEnding = Ds.Eof

Видалити поточний запис таблиці Delete()

Приклад:

'встановимо курсор на останній запис та видалимо його Ds.Last() Ds.Delete()

Очистити таблицю Clear(Full As Boolean)

Clear(Full As Boolean

Вхідні параметри:

Full - *true* - ознака повної очистки таблиці / *false* (за замовчуванням) - таблиця буде очищена та створено один порожній запис

Приклад:

Ds.Clear() Ds.Clear(true)

Отримати значення поля поточного запису. Повертає 0 якщо значення Null FldVal(FldName As String) As Variant

Вхідні параметри:

FldName - назва поля

Відповідь:

значення що міститься у полі поточного запису таблиці.

Приклад:

value = Ds.FldVal("Field1")

Отримати значення поля поточного запису. Повертає Null якщо значення Null FldOrigVal(FldName As String) As Variant

Вхідні параметри:

FldName - назва поля

Відповідь:

значення що міститься у полі поточного запису таблиці.

value = Ds.FldOrigVal("Field1")

Встановити/отримати поточне сортування записів таблиці

Sort As String

Сортування встановлюється в форматі : " «назва поля» ASC(DESC) "

Приклад:

Ds.Sort = "Field1 ASC, Field2 DESC"

Встановити/отримати значення фільтру відображення даних в таблиці Filter As String

Приклад:

Ds.Filter = "Field1 > 4 AND Field2 = 'in' AND Field3 Is Not Null"

***Зауваження.** Допускається послідовне накладання фільтрів для отримання окремих вибірок даних з таблиці із збереженням вихідного масиву даних.

Приклад:

Таблиця Ds містить дані: Field1 Field2 5 out 3 out 10 in 2 in 7 out Ds.Filter = "Field1 > 4 " 'Ds.MaxRecords = 3 Ds.Filter = "Field2 = 'in'" 'Ds.MaxRecords = 2 Ds.Filter = "Field1 > 4 AND Field2 = 'out' " 'Ds.MaxRecords = 2

Створити порожній об'єкт (таблицю) IZDataset

GetEmptyDataSet() As IZDataset

Відповідь:

IZDataset – порожня таблиця.

Приклад:

set App = CreateObject("Medoc.ZApplication")
set DS = App.GetEmptyDataSet()

2.9. IZFields - об'єкт колекція полів рядка таблиці

Отримати об'єкт поля запису таблиці

Item(Name As Variant) As IZField

Вхідні параметри:

Name - ім'я поля/індекс у запису таблиці

Відповідь: об'єкт поля <u>IZField</u>

Приклад:

Fld = Ds.Fields.Item(0) Fld = Ds.Fields.Item("Field1")

Кількість полів у запису таблиці Count As Long

Приклад:

cnt = Ds.Fields.Count

Додавання нового поля в колекцію полів IZDataset AddField(fieldName As String, fieldType As ZDataTypeEnum)

Вхідні параметри:

fieldName – назва поля; fieldType – тип поля з Перелік <u>ZDataTypeEnum</u>

Приклад:

set App = CreateObject("Medoc.ZApplication")
set Ds = App.GetEmptyDataSet()
Ds.Fields.AddField "TextField", 2

Перелік ZDataTypeEnum - типи даних поля:

- 0 type Long
- 1 type Float
- 2 type Text
- 3 type Bool
- 4 type Date
- 5 type Time
- 6 type DateTime

2.10. IZField - об'єкт поля рядка таблиці

Назва поля

Name As String

Приклад:

fldname = Ds.Fields.Item("Field1").Name

Отримати/встановити значення поля рядка таблиці

Value As Variant

Приклад:

fldvalue = Ds.Fields.Item(0).Value Ds.Fields.Item("Field1").Value = 123

Тип даних поля

Type As ZDataTypeEnum

Приклад:

fldtype = Ds.Fields.Item("Field1").Type

2.11. Додаткова інформація

Запуск М.Е. Оос з командної строки

Для station.exe та ezvit.exe - доступні наступні аргументи запуску:

-user : ім'я користувача;

-pass : пароль (необов'язково);

-org : ЄДРПОУ організації;

-dept : філія (необов'язково);

Приклад використання в 1С:

ЗапуститьПриложение(«C:\Medoc\ezvit.exe -user Admin -pass 111 -org 11223344»); //буде виконано авторизацію під користувачем Admin з паролем 111 та відкриття організації з ЄДРПОУ 11223344

Отримання ідентифікатора документа ExDocID в 1C

Довжина **ExDocID** у М.Е.Doc обмежена **38** символами. ІД у 1С 8.2 - **36** символів, що дає можливість використовувати додаткові 2 символи (наприклад, додаючи необхідний префікс або суфікс для своїх цілей до оригінального ІД).

Значення ExDocID записується у поле бази даних (CARD.ExDocID)

Версія 7.7:

//створимо функцію, що отримує об'єкт та повертає його ІД: Функция ПолучитьИД(Объект) Перем Стр; Стр = ЗначениеВСтрокуВнутр(Объект); СЗ = СоздатьОбъект("СписокЗначений"); СЗ.ИзСтрокиСРазделителями(Сред(Стр,2,СтрДлина(Стр)-2)); Возврат СЗ.ПолучитьЗначение(СЗ.РазмерСписка()); КонецФункции //приклад: «Список» – назва таблиці відібраних документів певного типу. Перебираємо рядки таблиці циклом, задаємо параметри: ВыбДок = Список.Документ; ВыбБланк = "J1201010"; PDate=Дата(ДатаГод(Текущаядата()),ДатаМесяц(ВыбДок.ДатаДок),01); ИД = СокрЛП(ПолучитьИД(ВыбДок)); Сообщить("Отримали ІД документа = " + ИД); Doc = App.OpenOrCreateByExDocID(ВыбБланк, 0, PDate, , ИД, 0);

Версія 8.2:

//тут більш проста ситуація з ІД. Спочатку визначаємо масив:

МассивДокументов = Новый Массив;

Для каждого СтрокаТаблицыДокументов Из СтрокиТаблицыДокументов

Цикл

МассивДокументов.Добавить(СтрокаТаблицыДо кументов.Документ);

КонецЦикла;

//переберемо масив документів та знайдемо ІД:

ВыбБланк = "J1201010";

PDate=Дата(Год(Текущаядата()),Месяц(Докумен т.Дата),01);

ИД = Строка(Документ.УникальныйИдентификатор());

Сообщить("Отримали ІД документа = " + ИД);

Doc = App.OpenOrCreateByExDocID(ВыбБланк, 0, PDate, ,ИД, 0);

//у результаті – передали у М.Е.Doc документ з певним ІД. Тепер його можна перевіряти на наявність у М.Е.Doc (при спробі повторної передачі), дізнаватися статус відправки, переміщувати у Корзину, відправляти.

Вказівник

AcceptDoc – функції AFTER_SPR – внутрішні програми IZDataSet – інтерфейси docType – типи

ZPeriodTypEnum – переліки

AcceptDoc, 41 Add, 41 AddAppendix, 27 AddDocID, 41 AddField, 61 AFTER_SPR, 13 Append, 57

В

A

BEFORE_SPR, 13 **Bof**, 58

С

CARD.STATUS, 36 CardCode, 55 CertToCache, 50 Clear, 42, 44 ClearAppendix, 28 Count, 60 CreateDocumentByParam, 16 CreateGovQryByCharCode, 21 CreateNewRnn, 19 CreateOrOpenCertReq, 22

D

DataSets, 54 Delete, 58 DisableScripts, 56 docType, 26 DocumentsDataSet, 49

Ε

Edit, 57 EnableScripts, 56 Eof, 58

F

Fields, 57 FillNNStatusDS, 39 Filter, 60 First, 57 FldOrigVal, 59 FldVal, 59

G

GetAppendixSign, 29 **GetCurrDept**, 48 **GetCurrEdrpou**, 48 GetDocCrtDate, 36 GetDocKVT, 31 GetDocProtocol, 33 GetDocSign, 30 GetDocSignInfo, 30 GetDocStatus, 36 GetDocStatusByCode, 36 GetDocWebAdr, 51 GetEmptyDataSet, 60 GetGenID, 44 **GetInTrashStatus**, 35 GetInTrashStatusByCode, 35 **GetNaklRegDate**, 37 GetNaklRegNum, 37 **GetNNResponse**, 37 GetNNResponseList, 39 GetNNResponseNN, 38 GetPacketDoc().Add, 17 **GetPrgVersion**, 47 **GetPrimaryReestr**, 25 **GetRnnPortion**, 20 GetSendSTTByCode, 34 GetSendSTTByExDocID, 33 **GetTemplate**, 46 **GetTemplates**, 45

I

IDParent, 55 IsReadOnly, 43 ISubSysFilter.GetFilter, 53 Item, 60 IZDataSet, 14 IZDocument, 14 IZField, 15 IZFields, 15

L

Last, 58

Μ

MODIFY, 12 **MoveToTrash**, 34 **MoveToTrashByCode**, 34

Ν

Name, 61 **Next**, 58

0

Open, 43 OPEN, 12 OpenDocumentByCode, 23 OpenDocumentByExDocID, 24 OpenOrCreateByExDocID, 15 OpenOrCreateDocument, 18 OperType, 32

Ρ

PrintDocPDF, 28 Prior, 58

R

ReceiveCorrespondence, 49 RecNo, 57 RecordCount, 57 RefreshCertReq, 22 RefreshRstDoc, 54 RSTTYPE, 20, 22 RunModule, 52

Save, 45, 55 SAVE, 12 SeekOrg, 47 SelectFirm, 48 Send, 42 Sender, 27 SentGetNDSPLTSTT, 50 ShowDocument, 52 ShowDocumentByCode, 52 Sort, 59 Specif, 55 Status, 27

Type, 61

Value, 61

ZApplication, 14 ZPeriodTypEnum, 20

S

Т

V

Ζ