



Інструкція по СОМ-об'єктам

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

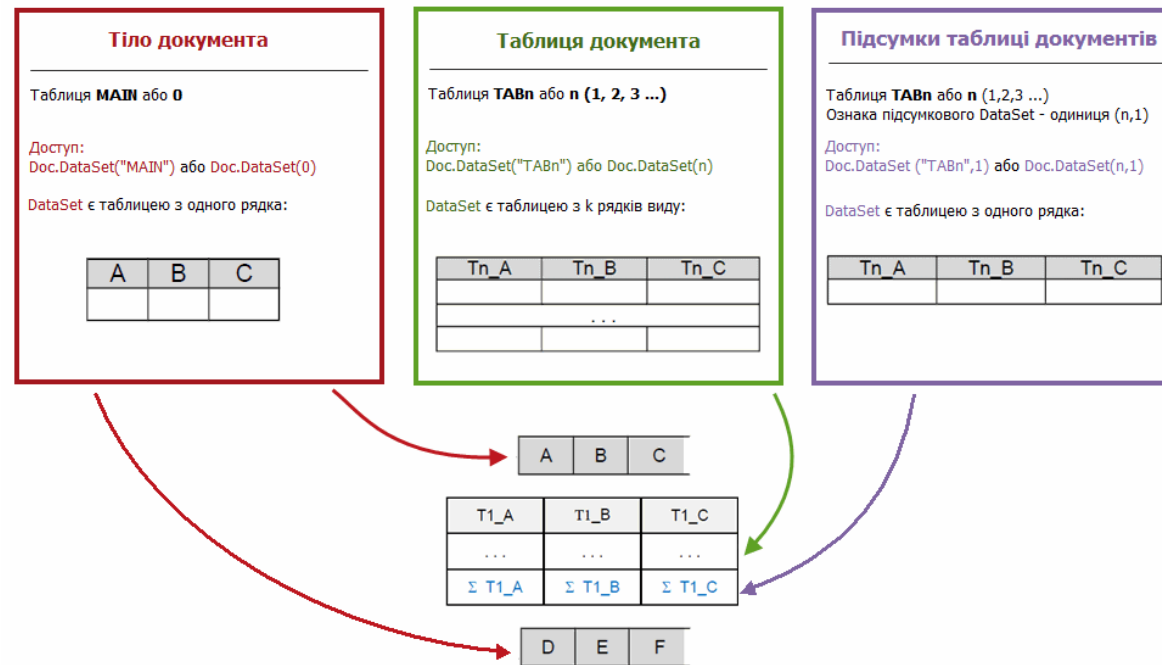
У інструкції також наводяться приклади використання методів програми М.Е.Дос, що можуть бути корисні під час роботи. У прикладах використовується мова 1С та VBS.

Зміст

1. Опис документів та VBScript програми документа	2
1.1. Структура документа	2
1.2. Імена полів у документі	3
Перегляд імен полів для будь-яких документів, крім первинних документів, що розроблені за власною структурою (шаблоном)	3
Перегляд імен полів документів, що розроблені за власною структурою (шаблоном).....	7
1.3. VBScript – програми документа	8
Приклади використання вкладки шаблону Програми	9
«CREATE»	11
«OPEN»	12
«SAVE»	13
«MODIFY»	14
«BEFORE_SPR»	15
«AFTER_SPR»	16
2. Робота з документами	17
2.1. Створення документів.....	18
2.2. Пошук інформації у базі з використанням ExDocID документа	25
2.3. Таблиці	34
2.4. Додаткова інформація	36
2.5. Отримання шаблону первинного документа	44

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.IZDataSets. 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.Дос Довідники – Реєстр бланків.
2. Знайдіть потрібний бланк у реєстрі та відкрийте бланк подвійним кліком мишею.
3. Ідентифікатори полів позначені аббревіатурами, що містять латинські символи.

Наприклад, код бланка S0301011, назва: 1-ПВ. Звіт з праці (місячна):



M.E.DOC

Файл

Правка

Вигляд

Сервіс

Довідка

Головне меню

Реєстр бланків

Перегляд бланка S0301011

Розділ І. Кількість працівників, робочий час і фонд оплати праці

Назва показників	Код рядка	За звітний місяць
А	Б	1
Фонд оплати праці усіх працівників, тис.грн. <i>(з одним десятковим знаком)</i>	1020	A1020
Сума податку з доходів фізичних осіб, що відрахована з фонду оплати праці усіх працівників (крім тимчасової непрацездатності), тис. грн. <i>(з одним десятковим знаком)</i>	1030	A1030
Середньооблікова кількість штатних працівників, осіб <i>(у цілих числах)</i>	1040	A1040
Кількість відпрацьованого робочого часу штатними працівниками, люд.год <i>(у цілих числах)</i>	1060	A1060
Фонд оплати праці штатних працівників, тис.грн. <i>(з одним десятковим знаком)</i> <i>(із ряд. 1020)</i>	1070	A1070

Пояснення до розділу І

З питань супроводження «М.Е.Дос» звертайтеся за телефоном: «До Розробника» - (044)206-72-10

Інструкція по СОМ-об'єктам

Для документів, які доступні у інших розділах програми:

1. Оберіть розділ програми М.Е.Дос, 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:

Таблиці документа	Ідентифікатор поля	Назва поля
Головна таблиця - [MAIN]	[A110]	Всього по розділу обсяг продажу без ПДВ
Таблиця - [TAB1]	[A111]	Загальна сума коштів
	[A17]	Всього по розділу обсяг продажу без ПДВ (20%)
	[A18]	Всього по розділу обсяг продажу без ПДВ (0%)
	[A19]	Всього по розділу обсяг продажу без ПДВ (0%)експорт
	[A2_10]	ТТВ-Обсяг продажу без ПДВ
	[A2_11]	ТТВ-Загальна сума коштів
	[A2_4]	ТТВ-одиниця виміру товару
	[A2_5]	ТТВ-Кількість
	[A2_6]	ТТВ-Ціна продажу
	[A2_7]	ТТВ-Обсяг продажу без ПДВ (20%)
	[A2_8]	ТТВ-Обсяг продажу без ПДВ (0%)
	[A2_9]	ТТВ-Обсяг продажу без ПДВ (0%)експорт
	[A3_11]	ЗЗТ-загальна сума коштів
	[A4_10]	Надано (знижка)-Обсяг продажу без ПДВ
	[A4_101]	Надано (знижка)-Обсяг продажу без ПДВ
	[A4_11]	Надано (знижка)-Загальна сума коштів

Інструкція по СОМ-об'єктам

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

The screenshot shows the 'me.doc' application interface. The top toolbar contains several icons, with the 'Показати бланк' (Show Blank) icon highlighted by a red rectangle. Below the toolbar, the main window displays a form for a tax invoice. The form includes fields for 'Зведена податкова накладна' (Summary tax invoice), 'Складена на операції, звільнені від оподаткування' (Compiled for operations exempt from taxation), and 'Не підлягає наданню отримувачу (покупцю)' (Not subject to provision to the recipient (buyer)). The form also includes fields for 'Податкові зобов'язання*' (Tax liabilities*), 'Податковий кредит*' (Tax credit*), and 'Кас. метод*' (Cash method*). The bottom right corner shows the 'ПОДАТКОВА НАКЛАДНА' (Tax Invoice) and the date '27122017'.

Відкриється шаблон документа, що містить ідентифікатори полів:

The screenshot shows the 'me.doc' application interface with the 'Перегляд бланка J1201009' (View Blank J1201009) window open. The window displays a form for a tax invoice. The form includes fields for 'Зведена податкова накладна' (Summary tax invoice), 'Складена на операції, звільнені від оподаткування' (Compiled for operations exempt from taxation), and 'Не підлягає наданню отримувачу (покупцю)' (Not subject to provision to the recipient (buyer)). The form also includes fields for 'Податкові зобов'язання*' (Tax liabilities*), 'Податковий кредит*' (Tax credit*), and 'Кас. метод*' (Cash method*). The bottom right corner shows the 'ПОДАТКОВА НАКЛАДНА' (Tax Invoice) and the date 'N11'.

Зведена податкова накладна	N25
Складена на операції, звільнені від оподаткування	N26
Не підлягає наданню отримувачу (покупцю)	N13
(вказується відповідний тип причини)	N14 N14

Податкові зобов'язання* PZOB
Податковий кредит* PKRE
Кас. метод* N24 D

ПОДАТКОВА НАКЛАДНА N11
Дата складання

Постачальник (продавець)

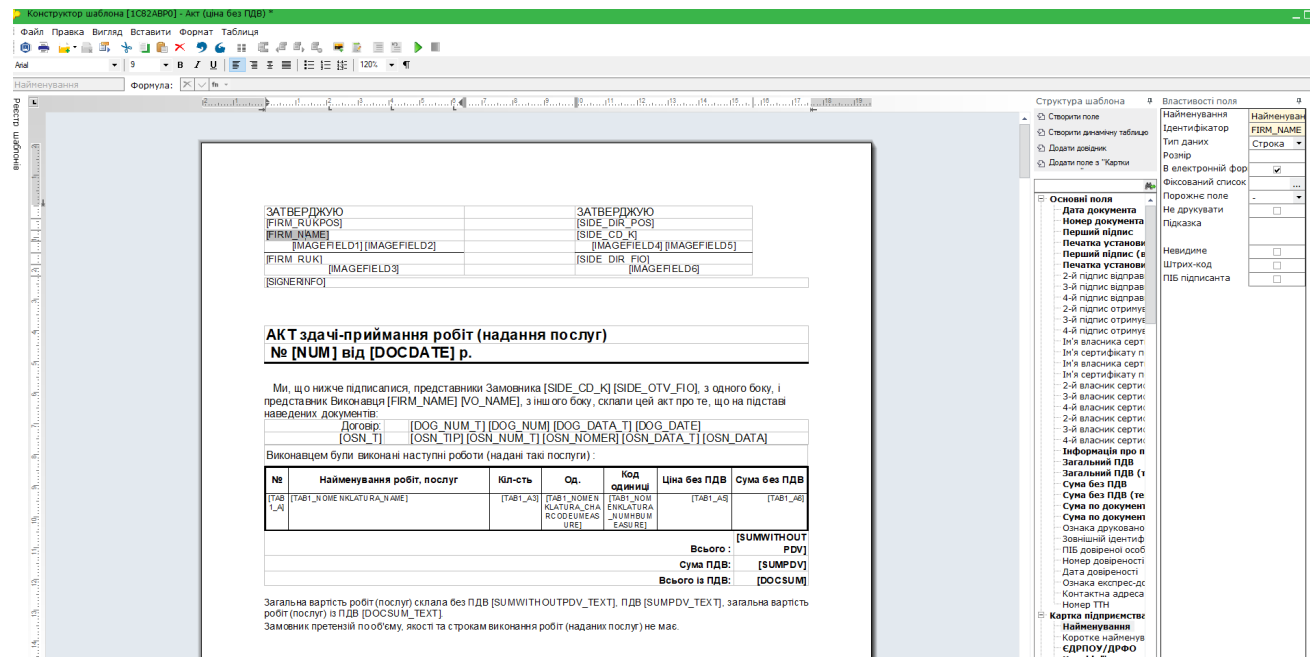
FIRM_NAME (найменування, прізвище, ім'я, по батькові - для фізичної особи - підприємця)	
FIRM_INN (індивідуальний податковий номер)	N2_13 (номер філії 2)

Отримувач (п

(індивідуальний

Перегляд імен полів документів, що розроблені за власною структурою (шаблоном)

1. Оберіть розділ програми М.Е.Дос Первинні документи – Шаплони первинних документів.
 2. У вікні ліворуч оберіть назву шаблону документа та відкрийте шаблон подвійним кліком мишею.
 3. У вікні редактора шаблонів імена полів, що можуть приймати певні значення, позначені ідентифікаторами у квадратних дужках.
- Властивості поля відкриваються кліком на певному полі.



Примітка:

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

1.3. VBScript – програми документа

Системні програми	Опис	Приклад
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 зміні будь-якого поля тіла документа. 2) Виконується при зміні 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 зміні 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.	
SAVE	Збереження. Виконується при збереженні документа	
Вкл / Вискл	Відключення / Включення 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	<code>Doc = App.OpenOrCreateByExDocID(Бланк,0,PDate, ,ІД,0); Doc.DisableScripts(); //відключаємо бейсік-скрипти</code>

Інструкція по СОМ-об'єктам

	на кількість, так як ціну бухгалтер корегує на копійку (для округлення), і потрібно передавати дані у М.Е.Дос «як є». Примітка: відключаються лише програми модифікації. Програма створення <code>Create()</code> таким чином не відключається.	<pre>rsMain=Doc.DataSets() ; //Створити документ rsMain.Edit(); //Редагувати //..... передати дані rsMain.Post(); Doc.EnableScripts();//включаємо скрипти Doc.Save(1); Doc.Close();</pre>
--	--	---

Програми за потреби прописуються на вкладці Програми шаблону. 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:

Акт здачі-приймання робіт (надання послуг)

№ [NUM] від [DOCDATE] [CITY]

Ми, що нижче підписалися, представник Замовника [SIDE_CD_K], з одного боку, і представник Виконавця [FIRM_NAME], з іншого боку, склали цей акт про те, що на підставі наведених документів:
Договір № [DOG_NUM] від [DOG_DATE]

Виконавцем були виконані наступні роботи (надані такі послуги):

№	Код ДКПП	Найменування робіт, послуг	Кіль-ть	Од.	Ціна без ПДВ	Сума без ПДВ
TAB1 F1	TAB1 F7	TAB1 F2	TAB1 F3	TAB1 F4	TAB1 F5	TAB1 F6
Разом:					[SUMWITHOUTPDV]	
Сума ПДВ:					[SUMPDV]	
Всього із ПДВ:					[DOC SUM]	

Інструкція по СОМ-об'єктам

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):

Структура шаблону

Створити поле
Створити динамічну таблицю
Додати довідник
Додати поле з "Картки установи"

Основні поля

Дата документа

Номер документа

Перший підпис

Печатка установи

Перший підпис (відповідний)

Печатка установи (відповідна)

2-й підпис відправника

3-й підпис відправника

4-й підпис відправника

2-й підпис отримувача

3-й підпис отримувача

4-й підпис отримувача

Ім'я власника сертифікату відп. особи

Ім'я сертифікату печатки

Ім'я власника сертифікату відп. особи (відповіді)

Ім'я сертифікату печатки (відповідне)

2-й власник сертифікату відп. особи

3-й власник сертифікату відп. особи

4-й власник сертифікату відп. особи

2-й власник сертифікату відп. особи (відповіді)

3-й власник сертифікату відп. особи (відповіді)

4-й власник сертифікату відп. особи (відповіді)

Інформація про підписантів

Загальний ПДВ

Загальний ПДВ (текст)

Сума без ПДВ

Сума без ПДВ (текст)

Сума по документу

Сума по документу (текст)

Ознака друкованої форми

Зовнішній ідентифікатор

ПІБ довіреної особи

Номер довіреності

Дата довіреності

Ознака експрес-доставки

Контактна адреса контрагента

Номер ТТН

Картка підприємства

Контрагент (2)

Поля користувача

Місто

Посада керівника контрагента

Поля динамічної таблиці [ТАВ1]

Поле 1 таблиці ТАВ1

Документ

Номер договору	Найменування	Номер	Сума	ПДВ	Тип	Користувач	Найменування
НУ-ОJ-98754	ТОВ "ТЕСТ КОНТРАГЕНТ"	РН-789-789	120,00	20,00	Накладна	Admin	Видаткова накла...

Структура шаблону

Створити поле
Створити динамічну таблицю
Додати довідник
Додати поле з "Картки установи"

Основні поля

Картка підприємства

Контрагент (2)

Поля користувача

Увага!

В шаблоні дозволено використовувати самостійно створені поля. В структурі шаблону потрібно натиснути на пункт «Створити поле», автоматично створиться група «Поля користувача» яка буде містити новостворене поле. Але самостійно створені поля не будуть відображатися в «Реєстрі первинних документів». В реєстрі відображається інформація системних полів з груп «Основні поля», «Контрагент», «Картка підприємства».

	120,00	20,00
--	--------	-------

«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  
14         .Edit  
15  
16         If .FldVal("DOCDATE") < DateSerial(2018,1,1) Then  
17             .FldVal("VO_NAME") = "Іванов І.І."  
18             .FldVal("VO_POS") = "директор"  
19         Else  
20             .FldVal("VO_NAME") = "Петров П.П."  
21             .FldVal("VO_POS") = "головний бухгалтер"  
22         End If  
23  
24         .Post  
25     End With  
26 END SUB
```

«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  
6             msgbox ("Сьогоднішня дата менше 01.01.2018")  
7         else  
8             msgbox ("Сьогоднішня дата ,більша 01.01.2018")  
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 стовбцю 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
26    rsMain.FldVal("DOCSUM") = rsMain.FldVal("SUMWITHOUTPDV") + rsMain.FldVal("SUMPDV")
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
10      rsTab1.FldVal("TAB1_NOMENKLATURA_NAME") = CurrSprDS.FldVal("Name")
11      If CurrSprDS.FldVal("Price") <> 0 Then
12        rsTab1.FldOrigVal("TAB1_A5") = CurrSprDS.FldOrigVal("Price")
13      Else
14        rsTab1.FldVal("TAB1_A5") = Null
15      End if
16      '----- одиниці виміру
17      If Not isempty (CurrSprDS.Fields.Item("NUMHBUMESURE")) Then
18        Spr = App.GetSprValue("HBUMESURE", "NUM", currSprDS.FldVal("NUMHBUMESURE"), "NAME")
19        If Cstr(Spr) <> "" Then
20          rsTab1.FldVal("TAB1_OV_NAME") = Spr
21        Else
22          rsTab1.FldVal("TAB1_OV_NAME") = ""
23        End if
24      End if
25      '-----
26    End If
27    rsTab1.Post
28    Set rsTab1 = Nothing
29
30  End With
31 END SUB

```

2. Робота з документами

Будь-яка VbScript-програма за замовчуванням у якості public змінних отримує 3 об'єкта:

Об'єкт	Опис	Приклад (використання у 1С)
App	Поточна програма: MEDOC.ZApplication	Версія 7.7: App=СтворитиОб'єкт("MEDOC.ZApplication"); Версія 8.2: App=Новий СОМОб'єкт("MEDOC.ZApplication"); Примітка: обов'язково створюється першим. Рекомендується перевірити підключення і обробити виключення. Без створення цього об'єкту подальша робота неможлива.
Doc	Поточний документ: MEDOC.IZDocument	//створення документа (крім первинних) Doc = App.CreateDocumentByParam(+параметри); //створення первинних документів, у т.ч. за власними шаблонами Doc = pd.CreateDocument(+параметри); Примітка: попередньо для первинних документів необхідно створити: orgId = App.SeekOrg(+ параметри); pd= App.GetPrimaryDocs();
rsMain	Таблиця полів тіла документа MEDOC.IZDataSets Еквівалентний конструкції: Set Main = Doc.DataSets()	rsMain=Doc.DataSets(); Примітка: використовується одразу після створення документа Doc (див.вище)

2.1. Створення документів

Об'єкти документа:

[MEDOC.ZApplication](#) – доступ до БД, спискам документів, документам, сервісні функції.

[MEDOC.IZDocument](#) – доступ до даних документа (основна таблиця, підтаблиці тощо)

[MEDOC.IZDataSets](#) – доступ до таблиці

[MEDOC.IZFields](#) – доступ до таблиці полів [MEDOC.IZDataSets](#)

Створення документів з використанням зовнішнього унікального коду документа облікової системи (ExDocID) Рекомендується при інтеграції з обліковими системами. Дозволить застосовувати великий набір функцій, що використовують ІД.		
Функція	Опис	Приклад використання у 1С
OpenOrCreateByExDocID (string CharCode, int PeriodType, date CrtDocDate, [string PartName], string ExDocID, bool rewrite)	Створення документів розділу "Реєстр звітів" та ПН/ПК Створює документ з ІД своєї облікової системи. Знадобиться надалі для інших функцій. Параметри: CharCode = код бланка; PeriodType = перерахунок PeriodTyp * CrtDocDate = PeriodDate (01 число місяця звіту); PartName – рекомендується НЕ вказувати; ExDocID – зовнішній ІД; rewrite – перезапис так/ні (1/0) по співпадінню ІД.	Doc = App.OpenOrCreateByExDocID("FJ1201009 ", 12, PDate, , IDDoc, 0); //Передаємо: Передані дані: FJ1201009 – код бланка; 12 – місяць року (Грудень); 01.12.2017 – (перше число); 1410184549 – зовнішній ідентифікатор документа у системі; 0 – ознака того, що документ передається вперше. а) ІД документа із своєї облікової системи; б) rewrite - булевий тип перезапису, якщо такий ІД знайдений. //Примітка: попередньо рекомендується скористатися функцією SelectFirm()
OpenOrCreateDocument (long orgId, string charCode, [string num], [date docDate], string exDocID, bool rewrite)	Створення документів розділу "Реєстр первинних документів" Створює документ з ІД своєї облікової системи. Знадобиться надалі для інших функцій.	Doc= pd.OpenOrCreateDocument(IdOrg, "1C82ABP" , , , IDDoc, 0); //Передаємо: а) ІД облікової системи; б) rewrite - булевий тип перезапису, якщо такий ІД знайдено. //Приклад: App = Новий СОМОБ'єкт("MEDOC.ZApplication"); orgId = App.SeekOrg("ЗначениеЄДРПОУ/ДРФО", "ЗначениеФилиал");

<pre> CreateNewRnn (string CharCode, string ExDocID, int PeriodType, DateTime CrtDate) //Додатково //рекомендується перевірити // існування GetRnnPortion (int PeriodType, DateTime CrtDate, int RstType) </pre>	Створення "Реєстру податкових накладних" Попередньо рекомендується перевірити на дублювання у періоді за допомогою <code>GetRnnPortion ()</code> та, якщо такий існує, зробити Возврат, або змінити його вміст методом <code>OpenOrCreateByExDocID()</code>, вказавши ІД цього РПН. CharCode код форми РПН, наприклад «J1201508» PeriodType тип періоду CrtDate дата створення (01 число+ММ.РРРР) /----- ----- Повертає максимальний номер порції РПН, який існує, по типу реєстру у заданому періоді PeriodType див. примітку, пункт 1. CrtDate дата сотворення (01 число + ММ.РРРР) RstType див. значення з пункта 2.4 "Додаткова інформація"	<pre> pd= App.GetPrimaryDocs(); Doc= pd.OpenOrCreateDocument(IdOrg, "1C82ABP",,,, IDDoc, 0); Doc = CreateNewRnn("J1201508", «123», 0, Дата(01,06,2017)); //розширимо перевіркою GetRnnPortion() Перем КодДокумента; //можливі і інші варіанти отримати (або задати) код документа (наприклад, J1201506). Перем МойМесяц="червень", МойГод=2017; // можливі інші варіанти отримати (або задати) місяць і рік нашого звіту //-----зробимо функцію для автоотримання періоду----- ФункцияПолучитьПериод(МойМесяц, МойГод) Перем Период; Если МойМесяц ="січень" Тогда Период=Дата(01, 01, МойГод); ИначеЕсли //.....аналогічно для 12 місяців Возврат Период; КонецФункции //-----основний код----- ---- App = Новый СОМОбъект("MEDOC.ZApplication"); НомерПорции=Строка(Документ.HNP); МойПериод= ПолучитьПериод("червень", 2017); Если КодДокумента= "J1201508" Тогда ТипРННКод=14; ТипРНННазвание="Общий"; ИначеЕсли //можемо зробити перевірку на інші типи РПН КонецЕсли; МояПорцияВПериоде= App.GetRnnPortion(Емемесячный, МойПериод, ТипРННКод); Если МояПорцияВПериоде =НомерПорции Тогда Сообщить(ТипРНННазвание + " реєстр з номером порції " + НомерПорции + "у періоді" + МойМесяц + МойГод + "вже існує!"); Иначе Сообщить(ТипРНННазвание + " реєстр з номером порції " + НомерПорции + " Не існує! " + Символы.ПС + "Створимо його."); </pre>
--	---	---

Інструкція по СОМ-об'єктам

		<pre> Doc = App.CreateNewRnn(КодДокумента, МойПериод, ТипРННКод); КонецЕсли; </pre>
Інтерфейс IPacketDoc. Метод Add (string parentExDocID, string childExDocID)	Інтерфейс IPacketDoc для роботи з пакетами. У нього метод: Add(string parentExDocID, string childExDocID) повертає код результату додавання: «0» - ОК; «1» - головний документ не є батьківським; «2» - дочірній документ не визначений для пакету; «3» - дочірній документ не може бути доданий. Для отримання інтерфейсу використовується метод ZApplication.GetPacketDoc()	<pre> App = Новый СОМОбъект("MEDOC.ZApplication"); //-----створюємо головний документ----- PDate = Дата(2017,12,01); mainID = "4444"; packetDoc = App.GetPacketDoc(); Doc = App.OpenOrCreateByExDocID("J0200119", 0, PDate, , mainID, 0); rsMain = Doc.DataSets() ; rsMain.Edit(); rsMain.Append(); rsMain.Post(); Doc.Save(1); Doc.Close(); //-----створюємо дочірній документ----- childID = "5555"; Doc = App.OpenOrCreateByExDocID("J0215119", 0, PDate, , childID, 0); rsMain = Doc.DataSets(); rsMain.Edit(); rsMain.Append(); rsMain.Post(); Doc.Save(1); Doc.Close(); //-----спробуємо створити пакет----- КодВозврата = packetDoc.Add(mainID, childID); Если кодВозврата = 0 Тогда Сообщить("Додано звіт в пакет"); ИначеЕсли КодВозврата = 1 Тогда Сообщить("Помилка додавання у пакет: головний док. не є батьківським"); ИначеЕсли КодВозврата = 2 Тогда </pre>

Інструкція по СОМ-об'єктам

		Сообщить("Помилка додавання у пакет: дочірній док. не визначений для пакету"); Иначе Сообщить("Помилка додавання у пакет: дочірній док не може бути доданий"); //примітка: звичайно це означає, що раніше передавались документи з такими ж ІД (основний або додаток). Якщо створюється новий комплект (нова звітна, уточнююча), то рекомендується всі ІД пакету змінити, наприклад ІД =ІД + «NZ» КонецЕсли;
CreateGovQryByCharCode (string CharCode)	Створення документа інформаційної довідки «Запит щодо отримання відомостей з ЄРПН». У інформаційній довідці створюється запит.	App = Новый СОМОбъект("MEDOC.ZApplication"); Doc = App.CreateGovQryByCharCode("J1300104"); //Створення Запита rsMain = Doc.DataSets(); rsMain.Edit(); //...Заповнення полів документа rsMain.Post(); Doc.Save(); Doc.Close();
ZPeriodTypeEnum PeriodTyp (список зумовлених значень)	Код типу періоду. Існують зумовлені значення: 0 – місяць. Для первинних документів, завжди вказуйте 0 – місяць; 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.	Doc=App.OpenOrCreateByExDocID(ВыбБланк,0,PDate, ,ИД,0); //Примітка: Використовується зазвичай для передачі регламентованої звітності, періоди якої звичайно – квартал (або наростаючий підсумок). Рекомендується створити функцію, яка буде повертати код на основі Дати регламентованого звіту. Отримане значення використовувати при створенні документа. Для квартальних (не наростаючим підсумком!) можна програмно вказати значення 10.
OpenDocumentByCode (long Code)	Відкриття документа по ідентифікатору cardCode – генерується при створенні документа у програмі М.Е.Док	Doc=App. OpenDocumentByCode (123456789)
ZDataSets DataSets ([VARIANT Tbl], [Long Totals])	Отримати об'єкт структури документа: - тіло документа; - підтаблицю;	

	- підсумки підтаблиці. Tbl – що програма хоче отримувати. Приймає і строкові, і числові значення. Doc.DataSets("MAIN") або Doc.DataSets(0) – отримати тіло документа. Doc.DataSets("TAB1") або Doc.DataSets(1) – отримати підтаблицю 1. Значення за замовчуванням 0. Totals – вибір (значення за замовчуванням 0): «0» – власне підтаблиця; «1» – підсумки. rsMain=Doc.DataSets();	
Refresh()	Повний перерахунок документа, аналог натискання F5-F6	Doc.Refresh();
Save(bool check)	Зберегти документ перевірити на помилки (1 – так / 0 – ні) (не вказане значення = 0) При збереженні можна перевірити на помилки (аналог натискання F4 у документі).	Doc.Save(1);
GetPrimaryReestr (int docType, int docMove, date dateBegin, date dateEnd)	Метод отримання списку документів з реєстру первинних документів. Параметри: docType – тип документа; docMove – напрямок документа (0 – вхідний, 1 – вихідний); dateBegin – дата початку діапазону; dateEnd – дата кінця діапазону. OpenDocumentByExDocID(ExdocID) - функція відкриття документа з отриманого списку. Параметри: ExdocID – ідентифікатор документа Тип docMove int «-1» - всі;	Необхідно викликати об'єкт з заданими параметрами: App = CreateObject("Medoc.ZApplication") prim = App.GetPrimaryDocs() Reestr = prim.GetPrimaryReestr(docType, docMove, dateBegin, dateEnd) У відповідь повертається об'єкт таблиця з полями. <Тип документа> = Reestr.Fields.Item("Receptdate ").Value; ...<Номер документа> = Reestr.Fields.Item("Regdate ").Value; Список полів у таблиці Sender – Походження документа; CharCode – Код документа; CardCode – ідентифікатор документа у М.Е.Дос; Status – статус; Vatsum - сума ПДВ;

Інструкція по СОМ-об'єктам

	«1» - вихідні; «2» - вхідні). Тип документа docType int «-1» - усі; «10100» - Податкова накладна; «10101» - Додаток 2; «10103» - Рахунок; «10104» - Акт; «10105» - Первинний документ – Накладна; «10106» - Інші Документи; «10107» - Банківська виписка; «10108» - Довіреність; «10109» - Договір. Початок періоду - dateBegin Date; Кінець періоду - dateEnd Date.	Exdoc_ID – ідентифікатор документа у зовнішній системі; Regdate – дата реєстрації; Doc_id – ідентифікатор документа; Doc_type – тип документа; 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 – ІПН платника; Partner_EDRPOU – ЄДРПОУ контрагента; Partner_IPN – ІПН контрагента; OSOBFIO – відповідальна особа; SENDUSER – ккористувач, що відправив первинний документ; EXPDATE – дата експорту документа; NOTATION – примітка. Для ПН/РК GETTIME – Дата та час отримання документа контрагентом; APPROVE – інформація з вкладки «Затверджено» про підписанта, хто затвердив документ зі сторони контрагента. Тип Sender int «-1» - не визначено; «1» - ЄРПН; «2» - контрагент; «3» - контрагент (імпортировано із зовнішньої системи); «10» - копіювання у РПН; «11» - контрагент(прийнято з сервера ДФС). Тип Status int «0» - Готується (для первинних документів - Акт, Рахунок....) ; «1» - Відправлено в ДФС (для НН/РК); «2» - Доставлено в ДФС (для НН/РК); «3» - Відхилено в ДФС (для НН/РК); «4» - Прийнято в ДФС (для НН/РК); (Інших даних поля «Status» для ПН/РК не передбачено.) «5» - Відправлено контрагенту (для первинних документів - Акт, Рахунок ...);
--	---	---

Інструкція по СОМ-об'єктам

		«6» - Доставлено контрагенту (для первинних документів - Акт, Рахунок ...); «7» - Затверджено контрагентом (для первинних документів - Акт, Рахунок ...); «8» - Отриманий (для первинних документів - Акт, Рахунок ...); «9» - Відхилено контрагентом (для первинних документів - Акт, Рахунок ...). OpenDocumentByExDocID (ExdocID) - функція відкриття документа із отриманого списку. (Exdoc_ID із отриманої таблиці).
--	--	---

2.2. Пошук інформації у базі з використанням ExDocID документа

Функції, що використовують ExDocID		
Функція	Опис	Приклад використання у 1С
Получение ExDocID (для 1С)	ІД документа у 1С (так само, як у будь-якій системі, побудованій на принципах баз даних), існує. Необхідно його отримати та передати. Примітка: 1) Довжина ExDocID у М.Е.Дос обмежена 38 символами. ІД у інформаційній системі 1С 8.2 - 36 символів, що дає можливість використовувати додаткові 2 символа (наприклад, додаючи необхідний префікс або суфікс для своїх цілей до оригінального ІД). 2) Значення ІД записується у поле бази даних (CARD. ExDocID) та може бути прочитано методом GetDictionaries(); 3) Слідкуйте за унікальністю ІД у системі!	Версія 7.7: <pre>//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 = "J1201004"; PDate=Дата(ДатаГод(Текущаядата()),ДатаМесяц(ВыбДок.ДатаДок),01); ИД = СокрЛП(ПолучитьИД(ВыбДок)); Сообщить("Отримати ІД документа " + ИД); Doc=App.OpenOrCreateByExDocID(ВыбБланк,0,PDate, ,ИД,0);</pre> Версія 8.2: <pre>//тут більш проста ситуація з ІД. Спочатку визначаємо масив: МассивДокументов = Новый Массив; Для каждого СтрокаТаблицыДокументов Из СтрокиТаблицыДокументов Цикл МассивДокументов.Добавить(СтрокаТаблицыДокументов.Документ); КонецЦикла; //переберемо масив документів та знайдемо ІД: ВыбБланк = "J1201004"; PDate=Дата(Год(Текущаядата()),Месяц(Документ.Дата),01);</pre>

Інструкція по СОМ-об'єктам

		ИД = Строка(Документ.УникальныйИдентификатор()); Сообщить("Отримали ІД документа =" + ИД); Doc = App.OpenOrCreateByExDocID(ВыбБланк,0,PDate,,ИД,0); ----- ----- //у результаті – передали у М.Е.Дос документ з 1С, з певним ІД. Тепер його можна перевіряти на наявність М.Е.Дос (при спробі повторної передачі), дізнаватися статус відправки, переміщувати у Корзину, відправляти.
GetDockVT (string ExDocID, long Cardcode)	Отримання квитанції про реєстрацію у ДФС/від контрагента Необхідно передати ID документа, або код документа із М.Е.Дос. ExDocID – зовнішній ідентифікатор облікової; CardCode – ідентифікатор документа у М.Е.Дос, параметр опційний, якщо він заповнений, за ним здійснюється пошук квитанції.	Необхідно викликати об'єкт App = CreateObject("Medoc.ZApplication") prim = App.GetPrimaryDocs() KVT = prim.GetDockVT(ExDocID); Або KVT = prim.GetDockVT("",Cardcode); У відповідь повертається таблиця даних <дата получения> = KVT.Fields.Item("RECEPTDATE ").Value; <имя файла> = KVT.Fields.Item("FILE_NAME ").Value; ... <статус> = KVT.Fields.Item("STATUS").Value; Список полів у KVT, які можуть бути заповнені: OPERDATE - Дата операції; OPERTYPE - Тип операції; KVT_TEXT - Текст квитанції (не підписаної); KVT_SIGN - Текст квитанції з підписами; REGNUM - Реєстраційний номер; REASON - Причина відхилення; RECEPTDATE - Дата отримання документа ДФС; FILE_NAME - Ім'я файла; Результат: отримання заповненої таблиці з даними по квитанціям від ДФС /контрагента
GetSendSTTByExDocID (string ExDocID)	Повертає -1 якщо не знайдено документ або значення коду статусу: «0» - Готується; «1» - Підписано бухгалтером; «2» - Підписано директором; «3» - Підписано печаткою установи; «4» - Відправлено;	Рекомендується перевіряти документ на наявність перед перезаписом. Наприклад: Если App.GetSendSTTByExDocID(IDDoc)= -1; Сообщить("Документ не передавался у М.Е.DOC!"); Тогда //створюємо документ Иначе

Інструкція по СОМ-об'єктам

	«6» - Отримано повідомлення; «9» - Прийнято; «10» - Не прийнято; «11» - Помилка відправки; «12» - На підпис; «13» - Архів; «14» - Отримано; «15» - Імпортований; «16» - Відхилено контрагентом; «17» - Підписано співробітником; «19» - Доставлено контрагенту; «20» - Затверджено контрагентом; «32» - Доставка контрагенту неможлива.	Сообщить("Документ найдено М.Е.Doc!"); //в залежності від необхідності, можемо зробити Возврат або переміщення у Корзину существующего (див. функцію MoveToTrash(string ExDocID)), наприклад задавши Вопрос() користувачу або визначивши поведінку програмно, чи задавши налаштування користувацького інтерфейсу КонецЕсли; //Попередній знайдений документ по ІД, якщо він у Корзині – позбавляється ІД. Таким чином, у системі буде завжди тільки один документ з унікальним ІД (в межах організації).
GetSendSTTByCode (cardCode As Long)	Повертає -1 якщо не знайдено документ або значення коду статусу: «0» - Готується; «1» - Підписано бухгалтером; «2» - Підписано директором; «3» - Підписано печаткою установи; «4» - Відправлено; «6» - Отримано повідомлення; «9» - Прийнято; «10» - Не прийнято; «11» - Помилка відправки; «12» - На підпис; «13» - Архів; «14» - Отримано; «15» - Імпортований; «16» - Відхилено контрагентом; «17» - Підписано співробітником; «19» - Доставлено контрагенту; «20» - Затверджено контрагентом; «32» - Доставка контрагенту неможлива.	Працює за аналогією з GetSendSTTByExDocID, тільки замість ExDocID використовує cardCode App.GetSendSTTByCode(22)
MoveToTrash (string ExDocID)	Функція переміщення документа у корзину	Если App.GetSendSTTByExDocID(IDDoc)<>-1 Тогда App.MoveToTrash(IDDoc); Doc=App.OpenOrCreateByExDocID("J1201004",0,PDate,,IDDoc,1);

Інструкція по СОМ-об'єктам

		КонецЕсли;
MoveToTrashByCode (cardCode As Long)	Функція переміщення документа у корзину по ідентифікатору cardCode	App.MoveToTrashByCode(22) // де 22 значення cardCode
GetInTrashStatus (string ExDocID)	Функція отримання статусу документа у корзині. Повертає: true - у корзині; false – ні.	// доповнимо попередній приклад перевіркою: Если App.GetSendSTTByExDocID(IDDoc)<> -1 Тогда Если App.GetInTrashStatus(IDDoc)= 0 Тогда Сообщить ("Документ не у Корзині, але ми його туди перемістимо"); App.MoveToTrash(IDDoc); Сообщить ("Перемістили у Корзину"); Иначе Сообщить ("Документ вже у Корзині"); КонецЕсли; Иначе Сообщить ("Документа нема у програмі"); КонецЕсли;
GetInTrashStatusByCode (cardCode As Long)	Функція отримання статусу документа у корзині по ідентифікатору cardCode. Повертає: true - у корзині; false – ні.	// Працює за аналогією GetInTrashStatus, тільки замість ExDocID використовує cardCode App.GetInTrashStatusByCode(22) // де 22 значення cardCode
GetDocStatus (string ExDocID)	Повертає значення Card.Status по значенню ExDocID «0» - документ не знайдено; «1» - чернетка; «2» - перевірений; «3» - помилковий; «4» - імпортований; «5» - копія; «6» - відісланий; «7» - контроль відключено.	КодСтатуса=Строка(App.GetDocStatus(123)); Сообщить («Код статусу документа: » + КодСтатуса); //рекомендується написати функцію, що повертає назву статусу по коду, та використовувати її для виведення користувачу на екран (або інших цілей).
GetDocStatusByCode (cardCode As Long)	Повертає значення Card.Status по значенню cardCode	// Працює за аналогією GetDocStatus, тільки замість ExDocID використовує cardCode

Інструкція по СОМ-об'єктам

	«0» - документ не знайдено; «1» - чернетка; «2» - перевірений; «3» - помилковий; «4» - імпортований; «5» - копія; «6» - відісланий; «7» - контроль відключено.	<pre>App.GetDocStatusByCode(22) // де значення 22 – значення cardCode</pre>
<pre>GetDocSender() Його методи: Add(string ExDocID) Send(bool toDPA) AddDocId(string DocID) AcceptDoc (string DocID, bool Accept, string rejectedText)</pre>	Функція відправки документів. Send(bool toDPA) повертає кількість успішно відправлених документів toDPA, приймає значення: 1 (або true) – відправка у контролюючий орган; 0 (або false) – відправка контрагенту. AcceptDoc(string DocID, bool Accept, string rejectedText) погодження/відхилення документа, де DocID— ідентифікатор документа у М.Е.ДОС; Accept: 1/true – погодити документ, 0/false – відхилити документ. Для того щоб відправити контрагенту повідомлення про відхилення (або прийняття) документа, потрібно викликати Send(bool toDPA) з параметром false. Якщо можливий тільки один отримувач, програма відправить документ автоматично, якщо у списку декілька контролюючих органів — програма запитає, у який контролюючий орган потрібно відправити.	<pre>Sender = App.GetDocSender(); Для Инд = 1 По Список.КоличествоСтрок() Цикл //перебираємо перелік документів у таблиці Sender.Add(IDDoc); КонецЦикла; cnt= Sender.Send(1); //відправляємо у ДПС перелік із N документів Сообщить ("Відправлено " + cnt + " документів");</pre>
<pre>ShowDocument (string ExDocID)</pre>	Відкриває документ за зовнішнім ExDocID Якщо відкривається список документів, то у програмі М.Е.Дос вони будуть розміщені зліва направо у порядку виклику із 1С. Якщо необхідно документи впорядкувати –	<pre>App.ShowDocument("123");</pre>

Інструкція по СОМ-об'єктам

	впорядковуємо їх ІД у списку, і тільки потім відкриваємо (пройшовшись по списку).	
ShowDocumentByCode (cardCode As Long)	Відкриває документ за cardCode. Якщо відкривається список документів, то у програмі М.Е.Дос вони будуть розміщені зліва направо у порядку виклику із 1С. Якщо необхідно документи впорядкувати – впорядковуємо їх ІД у списку, і тільки потім відкриваємо (пройшовшись по списку)	App.ShowDocumentByCode(22)
GetDocCrtDate (string ExDocID)	Отримати дату створення документа (фактично – дата передачі документа із 1С у М.Е.Дос)	Сообщить («Документ був переданий: » + App.GetDocCrtDate("123"));
GetNaklRegNum (string ExDocID)	Отримати номер реєстрації ПН/ПК у ДФС	see App = CreateObject ("Medoc.ZApplication") set Prim = App.GetPrimaryDocs() msgbox Prim.GetNaklRegNum ("111222")
GetNaklRegDate (string ExDocID)	Отримати дату реєстрації ПН/ПК у ДФС	see App = CreateObject("Medoc.ZApplication") set Prim = App.GetPrimaryDocs() msgbox Prim. GetNaklRegDate ("111222")
GetNNResponse (string Exdocid Request, string DOCNUMRequest, string DOCIDNN)	Отримання «Витягу з ЄРПН» за ідентифікатором «Запиту в ЄРПН» Метод викликається для одного з трьох параметрів (два інших - пусті) Параметри: Exdocid Request – ідентифікатор документа запиту; string DOCNUMRequest – номер запиту; string DOCIDNN - ідентифікатор документа, на який відправлено запит.	Необхідно викликати об'єкт, задавши необхідні параметри: App = CreateObject("Medoc.ZApplication") ; govReq = App.GetGovRequest() ; Doc = govReq.GetNNresponse() ; Метод повертає у відповідь документ «Витяг» з заповненими полями. TAB1 = Doc.DataSets("TAB1") ; отримаємо таблицю із документа «Витяг», де присутня інформація про реєстрацію ПН
GetNNResponseNN (dataset NN) поля в NN (string NN_NUM1 string NN_NUM2 string NN_NUM3 date DOCDATE string DOCTYPE)	Функція дозволяє отримати останній витяг з ЄРПН за даними вхідних ПН/ПК Параметри: DOCTYPE – тип документа (ПН або РК): НН = 1; РК = 2;	set App = CreateObject("Medoc.ZApplication") set filterDS = App.GetEmptyDataSet() ' додати поле "NN_NUM1" тип Text 2 filterDS.Fields.AddField "NN_NUM1", 2 ' додати поле "DOCDATE" тип Date 4 filterDS.Fields.AddField "DOCDATE", 4 ' додати поле "IPN" тип Long 0

Інструкція по СОМ-об'єктам

	NN_NUM1 – номер накладної (порядковий номер); NN_NUM2 – номер накладної (вид діяльності); NN_NUM3 – номер накладної (номер філії); DOCDATE – дата документа.	<pre> filterDS.Fields.AddField "IPN", 2 ' додати поле "DOCTYPE" тип Long 0 filterDS.Fields.AddField "DOCTYPE", 0 'заповнюємо рядок filterDS.Edit() filterDS.Fields.Item("NN_NUM1") = "123" filterDS.Fields.Item("DOCDATE") = CDate("13.04.2018") filterDS.Fields.Item("IPN") = "150419931234" filterDS.Fields.Item("DOCTYPE") = 1 filterDS.Post() ' пошук витягу за списком параметрів set govReq = App.GetGovRequest() set DocReq = govReq.GetNNResponseNN(filterDS) set main = DocReq.DataSets(0) ' вивести результат WScript.Echo "ІПН підприємства = " & main.Fields.Item("FIRM_INN").Value WScript.Echo "№ запиту = " & main.Fields.Item("N1").Value </pre>
GetNNResponseList (date datebeg, date dateend, long num_beg, long num_end)	Функція повертає список «витягів» за діапазонами дат та інтервалами номерів. Якщо num_beg або num_end дорівнює 0, то він не враховується Datebeg – початкова дата діапазону для вибору «витягів»; date dateend – кінцева дата діапазону для вибору «витягів»; long num_beg – початковий номер «витягу»; long num_end – кінцевий номер «витягу»; Заповнюється datebeg і dateend і num_beg і num_end. Відповідь: dataset list(docid Response) список «витягів»	Необхідно викликати об'єкт з заданими параметрами: App = CreateObject("Medoc.ZApplication"); govReq = App.GetGovRequest() ResponseList = App.GetNNresponseList(datebeg,dateend, num_beg, num_end) Відповідь: DataSet із заповненими полями Поля у DataSet ResponseList : CARDCODE – ідентифікатор документа у М.Е.Дос, завжди заповнений REPNUM – номер запиту у ЄРПН REPDATE – дата запиту у ЄРПН(на дату) Після отримання списку «витягів», необхідна функція: OpenDocumentByCode (CARDCODE) – відкриття документа за ідентифікатором
Dataset FillINNStatusDS(Dataset filNNDS);	Перевірка наявності результату реєстрації по ПН/ПК Метод дозаповнює filNNDS інформацію про реєстрацію документів. Поля filNNDS:	Необхідно викликати об'єкт: filNNDSRez= App.FillINNStatusDS(filNNDS) У відповідь повертається дозаповнена таблиця даних з колонкою Status, REGDATE, VATSUM filNNDSRez. Fields.Item("VATSUM ").Value – сума ПДВ з документа

Інструкція по СОМ-об'єктам

	DOC_ID – ідентифікатор документа ExDocID; DOC_TYPE – тип документа, 1 – ПН, 2 –ПК; DOC_MOVE – 1 – видані, 2 – отримані; DOC_NUM – номер документа; DOC_VD - вид діяльності; DOC_DEPT - код філії; DOC_DATE – дата документа; FIRM_EDRPOU – ЄДРПОУ підприємство (для виданих – продавець, для отриманих – покупець); PARTNER_EDRPOU – ЄДРПОУ контрагента; PATRNER_IPN – ІПН контрагента; STATUS – статус; REGDATE – дата реєстрації; VATSUM - сума ПДВ; (Наприклад для пошуку документа можна передати наступні значення полів: DOC_ID, DOC_TYPE, DOC_MOVE, DOC_NUM) STATUS приймає значення: «-1» - відсутній документ; «-2» - критеріям пошуку відповідає більше одного документа; «-3» - відсутній обов'язковий параметр; «0» - не відправлялися на реєстрацію; «1» - відправлено у ДФС; «2» - доставлено у ДФС; «3» - не прийнято у ДФС; «4» - прийнято у ДФС.	FillNNDsRez. Fields.Item("REGDATE").Value – дата реєстрації FillNNDsRez. Fields.Item("STATUS ").Value – статус документа
Long AddAppendix (string fullFileName, string exDocID, long cardCode)	Додавання вкладення до первинного документа string fullFileName — абсолютний шлях до вкладення (для сервера, відносно сервера, де встановлений М.Е.Дос) Коди повернення: «0» - успішно; «1» - нема ідентифікатора документа; «2» - нема документа;	App = Новый СОМОбъект("MEDOC.ZApplication"); prim = App.GetPrimaryDocs(); a = prim.AddAppendix("C:\12.txt","",324235); //Додавання вкладення по кардкоду b=prim.AddAppendix("C:\12.txt","1b2671bd-a7d1-11e6-a5d9-ba249ecdad7a",0); //Додавання вкладення по ІД

Інструкція по СОМ-об'єктам

	«3» - для x32 обмеження на 50 вкладень, уже в наявності; «4» - обмеження на довжину імені файла, більше 50 символів; «5» - обмеження на загальний розмір 20М; «6» - для x32 обмеження на 50 вкладень з новим вкладенням; «7» - помилка додавання у базу; «8» - інші помилки, відсутність або блокування файла тощо.	
--	---	--

2.3. Таблиці

Робота з таблицями		
Функція	Опис	Приклад використання у 1С
<code>GetDictionary (string Name)</code>	Робота з довідниками. Приклади: HBPARTNER – контрагенти; HBGOODS – номенклатура; PHYSPERSONS – співробітники; HBTAXINSP- податкові інспекції; HBVO – відповідальні особи. Примітка: 1) Деякі довідники ведуться у розрізі організацій (контрагенти, співробітники, номенклатура, відповідальні особи). При роботі з такими довідниками рекомендується додати умову перевірки по полю IDORG, значення якого можна отримати функцією <code>App.SeekOrg("ЄДРПОУ", "філія")</code>; 2) Деякі довідники об'єднані групами (контрагенти, номенклатура). Попередньо необхідно створити групу та вказати її при створенні записів довідника, інакше вони не будуть відображені у програмі.	<pre>App = Новый СОМОбъект("MEDOC.ZApplication"); Dict = App.GetDictionary(); tabl = Dict.Open("HBPARTNER"); //отримати довідник контрагентів tabl.First(); Пока НЕ tabl.EOF Цикл Сообщить(tabl.Fields.Item("NUM").Value); //вивести поле NUM Сообщить(tabl.Fields.Item("NAME").Value); //вивести поле NAME ЕСЛИ tabl.Fields.Item("NUM").Value = "0" ТОГДА tabl.Remove(); //Видалити значення довідника у базі! tabl.Next(); //продовжити; КонецЕсли; //змінюємо поле довідника у базі tabl.Edit(); tabl.Fields.Item("NAME").Value = tabl.Fields.Item("NAME").Value + "_123"; tabl.Post(); tabl.Next(); КонецЦикла; //якщо довідник не лише для читання, то можна зберігати isReadOnly = Dict.IsReadOnly(); Если НЕ isReadOnly Тогда Dict.Save(); КонецЕсли; //очищуємо довідник. Загальні довідники очищуються повністю, а для тих що ведеться у розрізі організації - видаляються значення лише поточної організації: tabl 2= Dict.Open("HBPARTNERGRP"); Dict.Clear("HBPARTNERGRP"); Dict.Save();</pre>
<code>GetGenID (string Table)</code>	Отримати значення генератора для таблиці довідника.	<pre>Dict = App.GetDictionary(); NextCode = Dict.GetGenID("HBPARTNERGRP");</pre>

Інструкція по СОМ-об'єктам

	Повертає на 1 більше поточного. Одразу можна використовувати. Використовується для додавання коректної нумерації внутрішнього ІД записів. При додаванні записів у таблиці використання обов'язкове!	<code>Сообщить("Наступний код = " + NextCode);</code>
<code>ZFields Fields()</code>	Отримати колекцію об'єктів полів поточного запису. Для доступу до читання значень полів використовуйте <code>FldVal(...)</code> або <code>FldOrigVal(...)</code>	<code>//передаємо значення поля у М.Е.Дос, із аналогічного поля у 1С:</code> <code>rsMain.Fields.Item("NUM").Value = Док.Номер;</code> <code>//отримаємо значення поля з М.Е.Дос:</code> <code>num = rsMain. FldVal("NUM");</code>
<code>Append()</code>	Додати запис	<code>rsMain.Append();</code>
<code>Edit()</code>	Редагувати запис	<code>rsMain.Edit();</code>
<code>Post()</code>	Підтвердити зміни	<code>rsMain.Post();</code>
<code>Bool Bof</code>	Ознака початку таблиці	<code>tabl.BOF</code>
<code>Bool Eof</code>	Ознака кінця таблиці	<code>tabl.EOF</code>
<code>First()</code>	Встановити курсор на перший запис таблиці (зазвичай перед циклом проходження по таблиці)	<code>tabl.First();</code>
<code>Next()</code>	Перехід на наступний запис	<code>tabl.Next();</code>
<code>Prior()</code>	Перехід на попередній запис	<code>tabl.Prior();</code>
<code>Last()</code>	Перехід на останній запис	<code>tabl.Last();</code>
<code>String Sort</code>	Поточне сортування	<code>rsMain.Sort = "A ASC, B DESC"</code>
<code>String Filter</code>	Поточний фільтр	<code>rsMain.Filter = "A > 123 AND B < 124 AND C<DateValue('01.01.2001')"</code>
<code>Clear([Bool Full])</code>	Очистити таблицю. За замовчуванням <code>Full = false</code>	Если <code>Full=true</code> , видаляються всі записи, якщо <code>false</code> , - залишається один, очищений.
<code>Cancel()</code>	Відмовитись від змін	<code>tabl.Cancel();</code>
<code>Long RecNo</code>	Номер поточного запису	<code>tabl.RecNo();</code>
<code>RecordCount()</code>	Підрахувати кількість записів у таблиці	<code>abl.RecordCount();</code>

2.4. Додаткова інформація

Функція	Опис	Приклад використання у 1С
Запуск М.Е.Дос з параметрами	-user ім'я користувача; -pass пароль (необов'язково) -org ЄДРПОУ організації -dept філія (необов'язково)	<code>ЗапуститьПриложение(«C:\ProgramData\Medoc\Medoc\ezvit.exe - user Admin -org 12312300»);</code>
<code>GetPrgVersion()</code>	Отримати версію М.Е.Дос, з якою працюємо. Примітка: версія має текстовий формат з роздільником – пробіл. Тому для порівняння необхідно видалити пробіли або виділити декілька символів (пропонується останні 3).	<code>Попытка ВерсияМедка=App.GetPrgVersion(); ВерсияМедкаЧитабельная="10.00." + Прав(ВерсияМедка, 3); Если Число(Прав(ВерсияМедка, 3)) < 112 Тогда Сообщить ("Для коректної роботи необхідна версія не нижче 10.00.112" + Символы.ПС + "У Вас встановлена версія " + ВерсияМедкаЧитабельная + Символы.ПС + "Оновіть програму!"); КонецЕсли; Возврат; Исключение Сообщить("Помилка при визначенні версії"); КонецПопытки;</code>
<code>SelectFirm</code> (string EDRPOU, string DEPT)	Віртуально переходимо у потрібне підприємство (наприклад: у 1С вибрано «ПП Іванов», а М.Е.Дос – «ПП Петров»). Важливо для створення первинних документів, функції створення яких НЕ містять ІД поточного підприємства. Примітка: якщо у М.Е.Дос не передбачається введення більше 1 підприємства, цей метод можна не використовувати. І навпаки, - рекомендується при використанні кількох підприємств у базі М.Е.Дос. Рекомендується перед створенням документа функціями: <code>OpenOrCreateByExDocID()</code> <code>CreateNewRnn()</code>	<code>//Використовується для функцій створення документів, що не містять у параметрах коду організації. Це може призвести до того, що документи одного підприємства будуть передаватися у інше підприємство. Щоб уникнути такої ситуації: ОпределимПредприятие = App.SelectFirm(ЕДРПОУ, Филиал); Если ОпределитьПредприятие Тогда Doc = App.OpenOrCreateByExDocID(Бланк, 0, PDate , , ИД, 0); КонецЕсли;</code>

Інструкція по СОМ-об'єктам

	Додати код визначення підприємства, де задати значення потрібного підприємства: ЄДРПОУ та філії.	
GetCurrEdrpou () и GetCurrDept ()	Визначають ЄДРПОУ та філію поточної організації, у якій працює користувач. Рекомендується порівнювати, чи передаємо документ потрібному підприємству.	<pre>//Рекомендується використовувати у комплексі: CurrEDRPOU = App.GetCurrEdrpou(); CurrDept = App.GetCurrDept(); CurrAll= CurrEDRPOU + CurrDept; Организацияв1С=КодЄДРПОУ + ЗначениеФилиала; Попытка Если CurrAll <> Организацияв1С Тогда Сообщить ("Документи не можуть бути передані! У програмі =" + Организацияв1С + ", а у М.Е.Дос підприємство =" + CurrAll); //якщо не співпадають, не передаємо документи Возврат; КонецЕсли; Исключение Сообщить("Помилка при виборі фірми"); КонецПопытки; //Примітка: після створення методу SelectFirm() така перевірка вже неактуальна.</pre>
RunModule (string Module)	Запуск модуля. Якщо він відкритий – переміщення фокусу на вкладку. Module "DocRst" - Реєстр документів; "PrimaryDocsRstNew" - Реєстр первинних документів новий; "GovQry" - Інформаційна довідка; "AdmCert " - завантажені «Сертифікати». GetFilter() Отримати таблиці фільтру для підсистеми для заповнення параметрами фільтрації. Якщо для підсистеми не задано фільтр, то повернеться пуста таблиця. ApplyFilter()	<pre>//звичайний запуск модуля App.RunModule("DocRst"); //реєстр первинних документів App.RunModule("PrimaryDocsRstNew"); //реєстр первинних документів новий App.RunModule("GovQry"); //інформаційна довідка App.RunModule("AdmCert "); // документи від АЦСК «Сертифікати» // Приклад використання для реєстру звітів: App = 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 = "01.01.2018"; //за 2018</pre>

	Застосувати фільтр для підсистеми. Параметри фільтрації не є обов'язковими, тобто можна не заповнювати не всі параметри. Для нових Первинних документів DOCTYPE: «10100» - Податкова накладна; «10101» - Додаток 2; «10103» - Рахунок; «10104» - Акт; «10105» - Накладна; «10106» - Інші документи. ROUTE: «1» - Вхідний; «2» - Вихідний. SUBROUTE: «-3» - На опрацюванні; «-2» - Архів; «-1» - Корзина. Для Реєстру первинних документів SENDSTT може приймати значення: New – Нові; Imported – Імпортовані; Checked – Перевірені; Error – Помилкові; CtrlOff - Контроль вимкнено; ForEmplSign - На підпис співробітнику; ForBuhSign - На підпис бухгалтеру; ForDirSign - На підпис директору; ForStampSign - На підпис печаткою; Signed – Підписані; InAWay - В дорозі; Sent – Подані; Delivered – Доставлені; HasKvt1 - Отримана квитанція №1; Accepted – Прийняті; Rejected - Не прийняті; Expired - Квитанція прострочена;	<pre>filterTbl.Fields.Item("PERTYPE").Value = -1; //всі filterTbl.Fields.Item("PART").Value = 0; //Податкова filterTbl.Fields.Item("SENDSTT").Value = "Rejected"; //Не прийняті reestrFil.ApplyFilter(); //застосувати фільтр //Приклад використання для нового Реєстру первинних 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 = 0; //тип документа filterTbl.Fields.Item("PERFROM").Value = "01.01.2018"; //період з filterTbl.Fields.Item("PERTO").Value = "01.01.2018"; //період до filterTbl.Fields.Item("PARTNEREDRPOU").Value = "9912312611"; //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; //1 на опрацюванні reestrFil.ApplyFilter(); //Приклад використання для інформаційної довідки: App = Новый СОМОбъект("MEDOC.ZApplication"); reestrFil = App.RunModule("GovQry"); //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 = "01.07.2018"; //період reestrFil.ApplyFilter();</pre>
--	---	---

Інструкція по СОМ-об'єктам

	DeliveredToPdr - Доставлено (у підрозділ); Received – Отримано; Recycle – Корзина.	
RefreshRstDoc ()	Програмно «Оновлює» вікно з переліком документів у М.Е.Дос.	//передали документ. Тепер оновимо екран у М.Е.Дос. Інакше доведеться натискати однойменну кнопку, щоб побачити на екрані зміни App.RefreshRstDoc(); Сообщить("Документ переданий. Перейдіть у М.Е.Дос");
RefreshCertReq (string ID)	Програмно «Оновлює» вікно з переліком «Документів від АЦСК» (модуль «Сертифікати»), по ІД переданого документа у М.Е.Дос	//створили Заявки: Doc = App.CreateOrOpenCertReq(int reqType, bool isUSC, «123»); //тепер оновимо екран у М.Е.Дос. Інакше доведеться натискати однойменну кнопку, щоб побачити на екрані зміни App.RefreshCertReq(«123»); Сообщить("Документ переданий. Перейдіть у М.Е.Дос");
DocumentsDataSet (string CondString)	Отримати список документів за умовою Параметри: String CondString – рядок умови пошуку. Повинен мати стандартний синтаксис оператора SQL WHERE. Позвертає: Об'єкт таблиця документів за умовою	App.DocumentsDataSet("PerDate > DateValue('01.01.2001') AND CharCode = 'J1201004'")
DataSet GetEmptyDataSet () Метод створення пустої таблиці.	DataSet GetEmptyDataSet() Метод створення пустої таблиці. Необхідний для використання у інших методах	Необхідно викликати метод для створення таблиці: Ds= App.GetEmptyDataSet()
AddField (string fieldName, ZDataTypeEnum fieldType);	Повертає таблицю без колонок та без рядків. AddField (string fieldName, ZDataTypeEnum fieldType); Метод додавання поля у таблицю. Метод викликається при кожному додаванні нового поля. Типи полів: typeLong = 0; typeFloat =1; typeText = 2;	Після чого викликати метод додавання поля у таблицю. Ds. Fields.AddField("Field1", 1)

Інструкція по СОМ-об'єктам

	typeBool = 3; typeDate = 4; typeTime = 5; typeDateTime = 6.	
Створення «Реєстру податкових накладних»	Необхідно аналізувати період та тип реєстру. В залежності від цього записувати значення коду у поле RSTTYPE. Якщо це поля відсутнє у документі, неможливо буде сформулювати на його основі Декларацію та Додаток до неї!	до 1 січня 2011 0 ЗАГАЛЬНИЙ; 1 Переробного підприємства; 2 Сільськогосподарського підприємства; 3 Сільськогосподарського товаровиробника; 4 Спеціальний режим оподаткування; 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. Приклад: <pre> PDate=Дата(ДатаГод(Дата1),ДатаМісяц(Дата1),01); Если Дата(PDate)<'2013.03.01' Тогда rsMain.Fields.Item("RSTTYPE").Value = 9; Иначе rsMain.Fields.Item("RSTTYPE").Value = 14; КонецЕсли; </pre>

Інструкція по СОМ-об'єктам

Створення «ЄСВ»	Необхідно попередньо створити форму "Списку таблиць" E0400F01.	Приклад: <pre> App = Новый СОМОбъект("MEDOC.ZApplication"); Doc = App.CreateDocumentByParam("E0400F01", 0, Дата(2013,06,01)); rsMain=Doc.DataSets(); //Створити документ rsMain.Edit(); //...Заповнення полів rsMain.Post(); Doc.Save(); Doc.Close(); Doc = App.CreateDocumentByParam("E0405F01", 0, Дата(2013,06,01)); rsMain=Doc.DataSets(); //Створити документ rsMain.Edit(); //...Заповнення полів rsMain.Post(); Doc.Save(1); Doc.Close(); </pre>
App.ReceiveCorrespondence()	Виконує прийом документів та квитанцій із ДФС, СДО та пошти (аналог F9 у інтерфейсі)	<pre> set App = CreateObject("Medoc.ZApplication") set sign = App.GetDocSigning() rem передача підпису, сертифікату та пароля у кеш res = sign.CertToCache("D:\cert\11223344\11223344\11223344_C150428095107.cer", "D:\cert\11223344\11223344\11223344_C150428095107.ZS2", "123") res = App.ReceiveCorrespondence() rem отримує повідомлення WScript.Echo "Res is " & res rem повертає: true — якщо є нові повідомлення; false – якщо нових повідомлень нема. </pre>
string GetDocSign (int CardCode, string Exdocid, int SignType)	Повертає підписаний документ у BASE64 по ExDocID або CardCode ExDocID – зовнішній ідентифікатор облікової системи CardCode – ідентифікатор документа у М.Е.Дос Тип SignType int «1» - отримуємо XML без підписів; «2» - підписаний XML;	<pre> App = CreateObject("Medoc.ZApplication") sign = App.GetDocSigning() Пошук по CardCode f = sign.GetDocSign(123456, "", 2) або Пошук по ExDocID f = sign.GetDocSign(0, "FLDKFJHS", 2) </pre>

Інструкція по СОМ-об'єктам

	«3» - підписаний XML, так як він зберігається у М.Е.Дос; «4» - підписаний PDF документ (первинні документи крім ПН та РК).	
DataSet GetDocSignInfo (int CardCode, string Exdocid)	Отримання підписантів документа ExDocID – зовнішній ідентифікатор облікової системи CardCode – ідентифікатор документа у М.Е.Дос Повертає таблицю з підписантами документа	<pre>App = CreateObject("Medoc.ZApplication") sign = App.GetDocSigning() По CardCode rez = sign.GetDocSignInfo(12523) по ExDocID rez = sign. GetDocSignInfo(0,"GHFGTFGHDF") rez.Fields.Item("SIGNLEVEL ").Value – тип сертифікату: «-1» - «не визначено» «0» - «директор» «1» - «бухгалтер» «2» - «печатка» «3» - «співробітник» «5» - «адміністратор реєстрації» «7» - «фізична особа» SIGNLEVEL буде мати значення відмінне від «-1» лише для сертифікатів, що видані АЦСК «Україна» rez.Fields.Item("IPN ").Value – ІПН rez.Fields.Item("FIO").Value – кому видано сертифікат rez.Fields.Item("ISSUER").Value – Видавець сертифікату rez.Fields.Item("SERIAL").Value – ідентифікатор ключа rez.Fields.Item("DATEBEG").Value – дата початку дії сертифікату rez.Fields.Item("DATEEND").Value – дата закінчення сертифікату rez.Fields.Item("TITLE").Value – посада rez.Fields.Item("SN").Value – серійний номер сертифікату</pre>
App.GetDocSigning() CertToCache (cert, key, pass)	Запис у кеш програми інформації про секретний ключ, пароль та сертифікат CertToCache (cert, key, pass) cert— текстовий параметр, у якому необхідно вказати фізичний шлях до зберігання сертифікату та ім'я файла key -- текстовий параметр, у якому потрібно вказати фізичний шлях , де зберігається секретний ключ та ім'я файла	<pre>App = CreateObject("Medoc.ZApplication"); set sign = App.GetDocSigning() res = sign.CertToCache ("D:\cert\11223333_1122334456_BS150428095136.crt", "D:\key\11223333_1122334456_BS150428095136.ZS2", "12356") WScript.Echo "Result is " & res</pre>

	pass – текстовий параметр, у якому потрібно вказати пароль Метод також повертає коди помилок: 0» - вдалось успішно записати пару ключ-сертифікат у кеш; «1» - за вказаним шляхом не знайдено сертифікат; «2» - за вказаним шляхом не знайдено ключ; «3» - вказано неправильний пароль «4» - не співпадає пара ключ-сертифікат; «5» - вказаний сертифікат не знайдено у базі програми; «6» - вказаний сертифікат належить іншій організації.	
DataSet SentGetNDSPLTSTT ([in] IZDataset NDSPLTSTT)	«Перевірка платників ПДВ» Метод відправки запиту «Перевірка платників ПДВ» IPN – Індивідуальний номер платника податків; PERDATE – дата, на яку необхідна відповідь; STATUS : «0» – відправлено запит; «1» - платник; «2» – неплатник; Обов'язкові для заповнення поля: IPN – Індивідуальний номер платника податків; PERDATE – дата, на яку необхідна відповідь.	Необхідно викликати об'єкт : <pre>App = CreateObject("Medoc.ZApplication"); prim = App.GetPrimaryDocs() NDSPLTSTTREZ = prim.SentGetNDSPLTSTT(NDSPLTSTT);</pre> Метод дозаповнює список NDSPLTSTT значенням STATUS У відповідь повертається заповнена таблиця даних <pre><ІПН> = NDSPLTSTTREZ.Fields.Item("IPN").Value;</pre> ... <pre><Статус> = NDSPLTSTTREZ.Fields.Item("STATUS").Value;</pre> Якщо по даному контрагенту існує результат, то інформація додається у відповідь. Інакше, здійсниться відправка запиту у ДФС по заданим даним.

2.5. Отримання шаблону первинного документа

Створення документів з використанням зовнішнього унікального коду документа облікової системи (ExDocID) Рекомендується при інтеграції з обліковими системами. Дозволяє застосовувати широкий набір функцій, що використовують ІД.		
Функція	Опис	Приклад використання у 1С
<code>GetParts()</code>	Повертає таблицю RDPART	<pre> IPrim = App.GetPrimaryDocs(); parts = IPrim.GetParts(); Пока НЕ parts.EOF Цикл Сообщить("CODE: " + parts.Fields.Item("CODE").Value + " NAME: " + parts.Fields.Item("NAME").Value); parts.Next(); КонецЦикла; </pre>
<code>GetTemplates</code> (int IDPart, int IDGrp)	Повертає RDSYSTMPL та RDUSERTMPL, де IDPart значення із таблиці RDPART, IDGrp (зарезервовано) приймає значення 0	<pre> IPrim = App.GetPrimaryDocs(); // 7 - отримати шаблони Первинних документів templates = IPrim.GetTemplates(7, 0); Пока НЕ templates.EOF Цикл curfld = 0; Пока curfld < templates.Fields.Count Цикл Сообщить(templates.Fields.Item(curfld).Name + ": " + templates.Fields.Item(curfld).Value); curfld = curfld + 1; КонецЦикла; Сообщить("-----"); templates.Next(); КонецЦикла; </pre>
<code>GetTemplate</code> (int IDPart, string CharCode)	Повертає XML шаблону, де IDPart значення із таблиці RDPART, CharCode чаркод шаблону	<pre> 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 елемента - поточна. Если Чтение.ТипУзла = ТипУзлаXML.НачалоЭлемента И Чтение.Имя = "field" Тогда </pre>

```

ИмяУзла = Чтение.Имя;
Сообщить("--" + ИмяУзла);
// Атрибути елементів можна читати тільки, якщо поточна частина
- початок елемента
Пока Чтение.ПрочитатьАтрибут() Цикл
// Прочитати дані вузла атрибуту.
ТипУзла = Чтение.ТипУзла;
Имя = Чтение.Имя;
Значение = Чтение.Значение;
Сообщить( Имя + ": " + Значение );
КонецЦикла;
ИначеЕсли Чтение.ТипУзла = ТипУзлаXML.Текст Тогда
// У прикладі просто виводимо текст у вікно повідомлення.
Сообщить("Текст:" + Чтение.Значение);
ИначеЕсли Чтение.ТипУзла = ТипУзлаXML.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="Таблица 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="" />
<field name="TAB1_FIELD3" type="Double" caption="Поле 3 таб.
TAB1" maxLength="0" spr="" sprname="" />
<field name="TAB1_FIELD4" type="Double" caption="Поле 4 таб.
TAB1" maxLength="0" spr="" sprname="" />
<field name="TAB1_FIELD5" type="Double" caption="Поле 5 таб.
TAB1" maxLength="0" spr="" sprname="" />
</partshema>

```