

Uruchamianie integracji Comarch BPM z Comarch HRM

Uwaga

Tryby Integracji Comarch BPM i Comarch HRM

Istnieją trzy główne tryby integracji dla systemów Comarch BPM i Comarch HRM, wynikające z ich lokalizacji (lokalnie lub w chmurze).

1. W pełni Lokalny z dostępem Zewnętrznym: Oba systemy (Comarch BPM i Comarch HRM) są zainstalowane lokalnie (on-premise) w środowisku klienta. W tym wariancie, jeżeli wymagany jest dostęp do BPM z zewnątrz (np. dla użytkowników mobilnych lub zdalnych), adres URL instancji Comarch BPM musi być wystawiony na zewnątrz, ale komunikacja między systemami (BPM ↔ HRM) odbywa się wewnątrz sieci LAN.

2. W pełni Chmurowy (Cloud): Oba systemy (Comarch BPM i Comarch HRM) są hostowane w środowisku chmurowym. Komunikacja między nimi jest zarządzana w chmurze i odbywa się bez ingerencji w infrastrukturę lokalną klienta.

3. Hybrydowy (Lokalny BPM z HRM w Chmurze): Comarch BPM działa lokalnie, natomiast Comarch HRM jest w chmurze. W tym wariancie adres URL instancji Comarch BPM musi być wystawiony na zewnątrz, aby umożliwić HRM (działającemu w chmurze) poprawną komunikację zwrotną do lokalnej instancji BPM.

Kiedy zostały już wykonane następujące czynności:

1. **dodanie** w Comarch BPM (dawniej DMS) systemu współpracującego z Comarch BPM (dawniej DMS) w zakładce „**API**” w zakładce „**Ustawienia**” (zob. [API](#));

2. **konfiguracja konta pracownika** w zakładce „API” w ramach **karty pracownika** w zakładce „Struktura organizacyjna” (zob. [Zakładka API na karcie pracownika – integracja BPM z HRM](#))

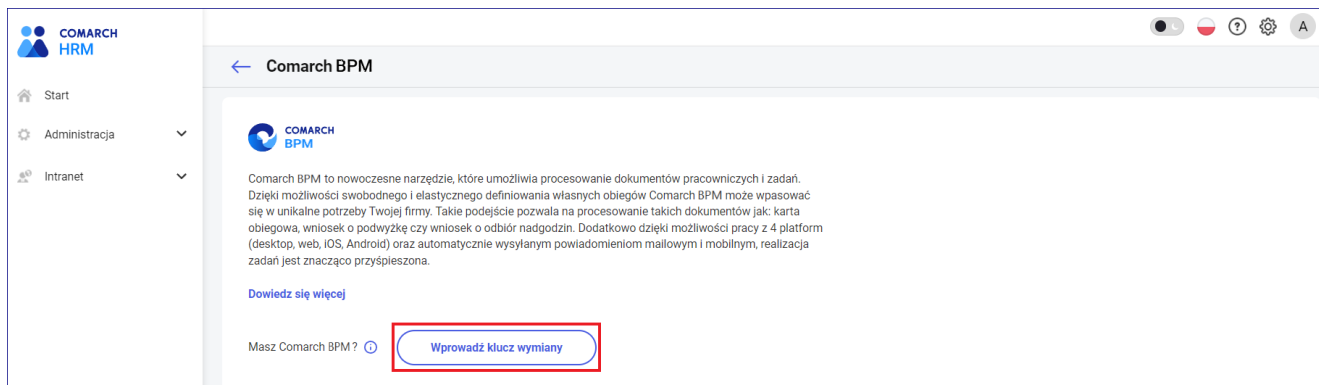
3. (od wersji 2024.2.2) **uzupełniono wartość klucza „WebClientUrl” w pliku Web.config** w katalogu z plikami aplikacji serwerowej (dawny web), **wpisując adres nowej aplikacji webowej Comarch BPM (dawniej DMS)** , a potem **zapisano zmiany** (zob. [Zmiany w integracji Comarch DMS z Comarch HRM](#))

wówczas administrator Comarch HRM może przystąpić do **uruchomienia integracji Comarch HRM z Comarch BPM (dawniej DMS)** .

W tym celu administrator Comarch HRM powinien:

1. **zalogować się** na stronę Comarch HRM;
2. **wybrać przycisk „Start”**;
3. w ramach pola *Włącz nowe funkcje* **kliknąć przycisk „Comarch DMS”**.

Następnie administrator powinien **nacisnąć przycisk *Wprowadź klucz wymiany***.



Przycisk „Wprowadź klucz wymiany”

Zostanie wówczas otwarte okno *Integracja z Comarch BPM*, w którym znajdują się dwa pola:

- **Adres WWW**
- **Klucz wymiany**


Integracja z Comarch BPM

Wprowadź adres WWW usługi oraz klucz wymiany

[Anuluj](#) [Zatwierdź](#)

Okno „Integracja z Comarch BPM”

Aby **poprawnie** uzupełnić pole **Adres WWW**, należy:

1. zalogować się do Comarch BPM (dawniej DMS) desktop jako administrator Comarch BPM (dawniej DMS) ;
2. wybrać spółkę **Comarch ERP Optima**
3. wybrać **kartę pracownika**, którego konta użyto do integracji z systemem współpracującym z Comarch HRM (zob. [Zakładka API na karcie pracownika – integracja BPM z HRM](#));
4. w zakładce **„API”** nacisnąć przycisk  **[Kopiuj URL aplikacji]**.

The screenshot displays the Comarch BPM 2025.2.0 interface. The top header shows 'Comarch BPM 2025.2.0' and '[Firma ABC] AS'. The main area is divided into three sections: 'Struktura organizacyjna' (Organizational Structure), 'Pracownicy' (Employees), and 'Pracownik' (Employee Details).

In the 'Struktura organizacyjna' section, the 'Pracownicy' tab is selected. A search bar contains 'Szukaj'. Below it, a list of employees is shown, with 'Anna Maj (AnnaMaj)' highlighted. A red box highlights the 'Pracownicy' tab and the 'Anna Maj (AnnaMaj)' entry.

The 'Pracownik' section shows details for 'Anna Maj (AnnaMaj)'. It includes tabs for 'OGÓLNE', 'UPRAWNIENIA', 'TYPY OBIEGU', and 'DANE OSOBOWE'. The 'DANE OSOBOWE' tab is active, showing the following information:

OGÓLNE	UPRAWNIENIA	TYPY OBIEGU	DANE OSOBOWE
System współpracujący	HR		
URL aplikacji	http://localhost/Stand/api/		
Klucz wymiany	XTYvJ4eDn9dzv2/D74y1whn6PC7XbNB+0U		

A red box highlights the 'API' label in the 'DANE OSOBOWE' tab. A tooltip 'Kopiuje URL aplik' is visible over the 'URL aplikacji' field.

Kopiowanie adresu URL aplikacji

Skopiowany w ten sposób **adres URL** należy **wkleić** w polu *Adres WWW* w oknie *Integracja z Comarch DMS* w Comarch HRM.

Integracja z Comarch BPM

Wprowadź adres WWW usługi oraz klucz wymiany

Adres WWW

http://localhost/Stand/api


Klucz wymiany

Anuluj

Zatwierdź

Wklejanie adresu URL skopiowanego w Comarch BPM (dawniej DMS) do okna w Comarch HRM

Analogicznie aby poprawnie uzupełnić pole *Klucz wymiany*, należy, będąc zalogowanym do Comarch BPM (dawniej DMS) desktop jako administrator Comarch BPM (dawniej DMS) :

1. wybrać spółkę **Comarch ERP Optima**;
2. wybrać **kartę pracownika**, którego konta użyto do integracji z systemem współpracującym z Comarch HRM (zob. [Zakładka API na karcie pracownika – integracja BPM z HRM](#));
3. w zakładce „**API**” nacisnąć przycisk  **[Kopiuj klucz wymiany]**.

Struktura organizacyjna Firma ABC

Pracownicy

Anna Maj (AnnaMaj)

Pracownik

OGÓLNE UPRAWNIENIA TYPY OBIEGU DANE OSOBOWE API

System współpracujący HR

URL aplikacji http://localhost/Stand/api/

Klucz wymiany XTYvJ4eDn9dzv2/D74y1whn6PC7XbNB+0U

Kopij klucz wymi

Kopiowanie klucza wymiany

Skopiowany w ten sposób **klucz wymiany** należy **wkleić** w polu *Klucz wymiany* w oknie *Integracja z Comarch BPM* w Comarch HRM.

Po wprowadzeniu poprawnych danych w polach *Adres WWW* i *Klucz*

wymiany należy **kliknąć** przycisk

Zatwierdź

[Zatwierdź].

Integracja z Comarch BPM

Wprowadź adres WWW usługi oraz klucz wymiany

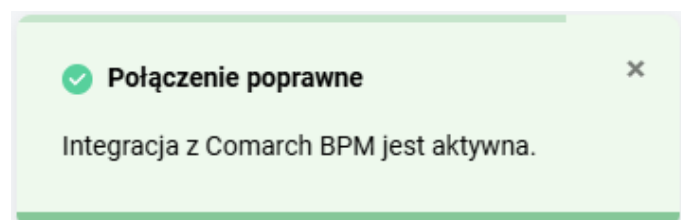
Adres WWW
http://localhost/Stand/api/

Klucz wymiany
.....

Anuluj **Zatwierdź**

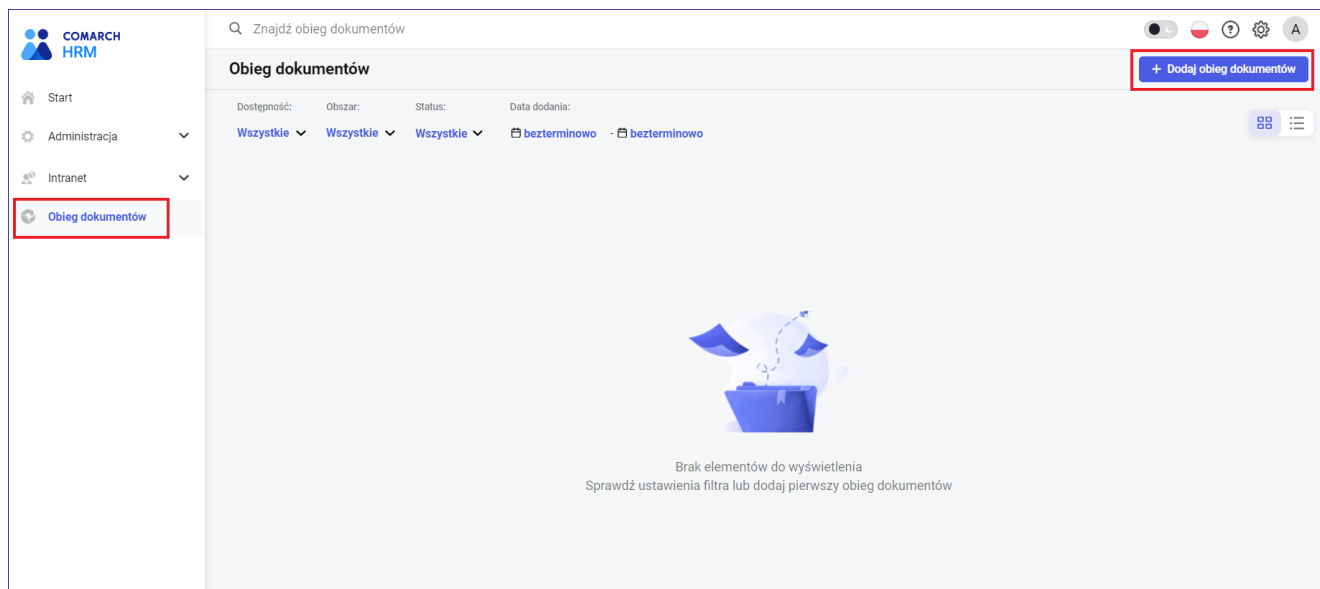
Wklejanie klucza wymiany skopiowanego z Comarch BPM (dawniej DMS) do okna w Comarch HRM i zatwierdzanie danych

Jeżeli uruchomienie integracji zakończyło się sukcesem, wówczas w prawym dolnym rogu strony zostanie wyświetlony następujący komunikat:



Komunikat informujący o poprawnym przeprowadzeniu integracji Comarch HRM z Comarch BPM (dawniej DMS)

Następnie administrator Comarch HRM w ramach zakładki „**Obieg dokumentów**” powinien **wybrać** przycisk ***Dodaj obieg dokumentów***.



Wybór przycisku „*Dodaj obieg dokumentów*”

Następnie administrator powinien **uzupełnić** następujące **pole**:

- **Nazwa** – w tym polu należy wpisać **nazwę**, jaką będzie nosił dany typ obiegu w Comarch HRM;
- **Obszar** – w tym polu można wybrać z rozwijanej listy **obszar** (domyślny obszar to „Ogólne”); Nowy obszar można dodać za pomocą przycisku ***Dodaj nowy obszar***;
- **Dostępność** – w tym polu można określić, **czy dany typ obiegu będzie dostępny dla wszystkich pracowników z Comarch ERP Optima** (domyślnie: „Publiczny”), czy tylko dla pracowników z określonego centrum w ramach struktury podległościowej. Jeśli wybrano centrum podległościowe, możliwe jest również **zaznaczenie** pod spodem parametru „Z podcentrami”
- **Proces** – pole, w którym z rozwijanej listy typów obiegu dostępnych w Comarch BPM (dawniej DMS) należy wybrać ten

typ obiegu, w ramach którego przez pracowników HRM będą tworzone nowe dokumenty

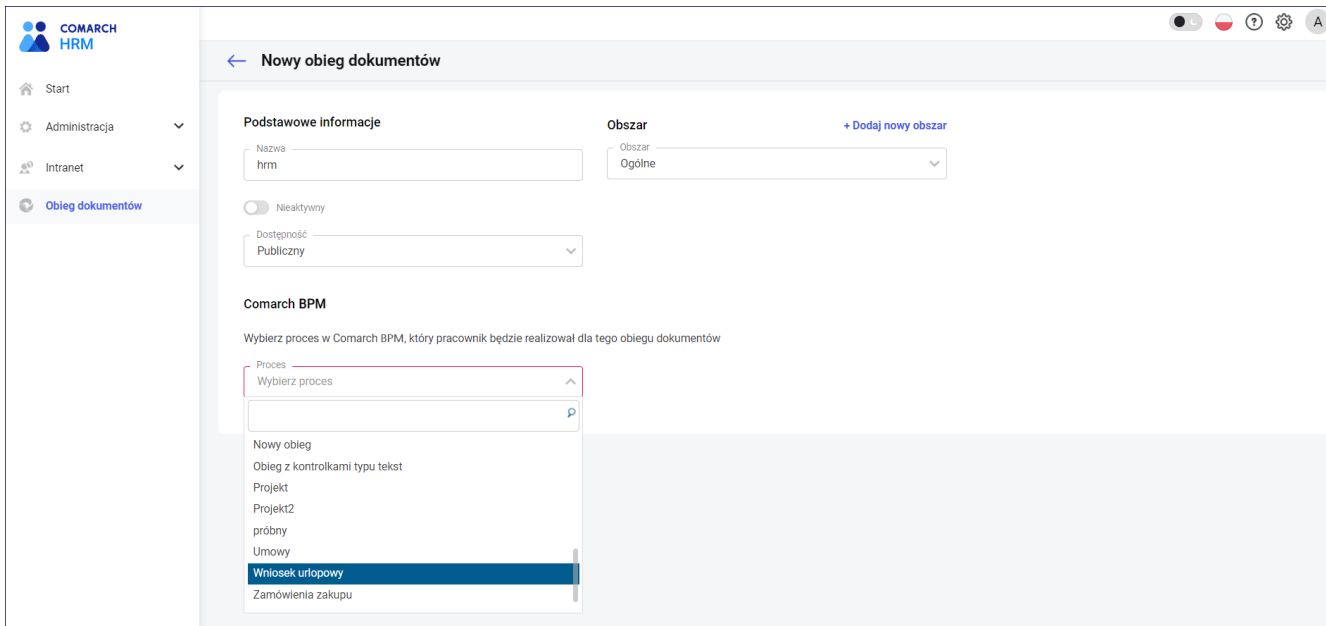
Obiegi dokumentów

Szukaj

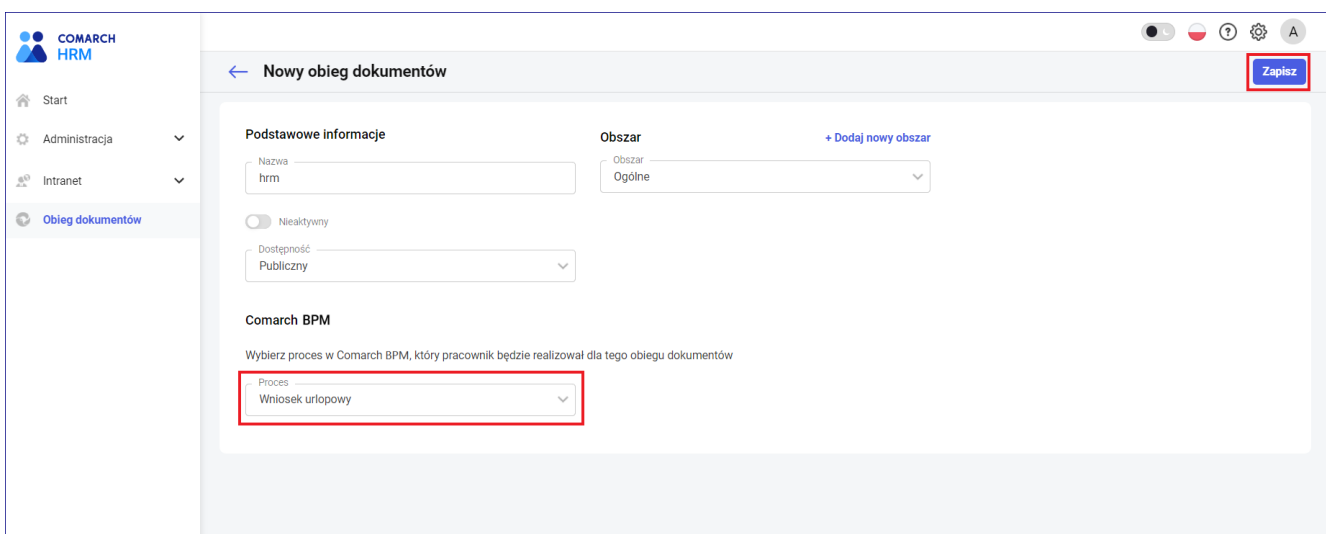
Nazwa	Prefix	Utworzono
Faktury zakupowe z XL	FZXL	03.10.2022 11:56:57
Faktury zakupu 2	FZ2	03.10.2022 14:10:45
Repozytoria		
Archiwum		10.10.2022 09:26:00
Umowy		10.10.2022 09:26:41
Archiwum2		18.10.2022 09:48:56
Faktury zakupu 3	FZ3	11.10.2022 08:20:45
Nowy obieg	NO	17.10.2022 08:44:13
próbny	pr	17.10.2022 09:50:21
Zmiany	Z	17.10.2022 15:04:06
Projekt2	PRO2	25.10.2022 13:32:20
Wniosek urlopowy	UR	25.10.2022 13:35:40
Faktury sprzedaży	SP	12.10.2022 10:53:38
Faktury zakupu 2022	FZ22	25.10.2022 13:39:24
Faktury zakupu ACD 2	FZACD	25.10.2022 13:46:39
Faktury		
Faktury zakupowe	FT	17.10.2022 08:49:43
Faktury zakupu 5	FZA	26.10.2022 08:39:19
Faktury - procedura do.OnAfterDocumentCreated	FP	31.10.2022 11:11:51
Blokada przekazania bez komentarza	BK	31.10.2022 12:07:08
Faktury zakupowe xl z listami	FL	08.11.2022 11:50:55
Obieg z kontrolkami typu tekst	OBO	09.11.2022 10:09:01
Faktury	FX	09.11.2022 13:13:14

Nazwa katalogu: Wszystkie obiegi

Lista typów obiegu dokumentów w ramach zakładki „Definicje obiegu dokumentów” w Comarch BPM (dawniej DMS) – zaznaczono typ obiegu „Wniosek urlopowy”




Wybór obiegu „Wniosek urlopowy” z Comarch BPM (dawniej DMS) w ramach Comarch HRM

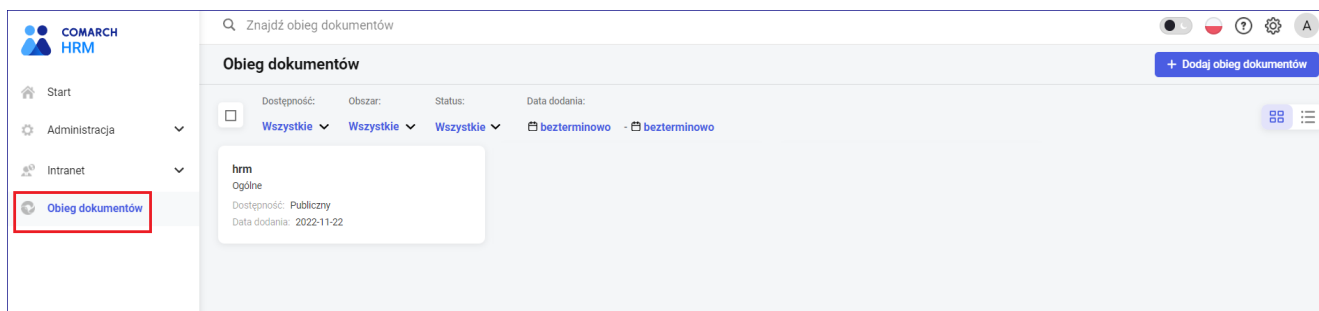


Wybór typu obiegu z Comarch BPM (dawniej DMS), z którym powiązany będzie obieg dokumentów określony w Comarch HRM i zapisywanie nowego obiegu dokumentów w Comarch HRM

Po **uzupełnieniu wszystkich pól** administrator powinien **klikać**

przycisk  [Zapisz].

Nowopowstały typ obiegu jest **widoczny** w Comarch HRM po kliknięciu na zakładkę **Obieg dokumentów** i jest dostępny dla pracowników, zgodnie z ustawieniami dostępności dla danego typu obiegu.



Widok typu obiegu dokumentów w Comarch HRM

Rozpoczynasz pracę z Comarch BPM (dawniej DMS) i chcesz dowiedzieć się, jak korzystać z programu? A może masz już podstawową wiedzę o Comarch BPM (dawniej DMS) i chcesz dowiedzieć się więcej?

[Sprawdź Szkolenia Comarch BPM!](#)

[Powrót do początku artykułu](#)

Czy wiesz, że kontrolkę typu Dane Tabełaryczne można automatycznie wypełnić pozycjami dokumentu zaczytanego przez Comarch OCR przy współpracy z Comarch ERP XL?

Zastosowanie

Dzięki zastosowaniu tej konfiguracji, **proces obiegu faktur kosztowych zyskuje pełną automatyzację**. Głównym celem jest eliminacja ręcznego wprowadzania pozycji dokumentów – dane zaczytane przez Comarch OCR są automatycznie mapowane i wypełniają kontrolkę tabelaryczną na kacie obiegu. W efekcie **skracamy czas obsługi faktury**, minimalizujemy błędy przepisywania i zapewniamy, że proces akceptacji bazuje na **poprawnych i spójnych danych** przed ich zaksięgowaniem.

Zawartość przykładu

- **Definicja Typu Obiegu (Workflow):** Wzorcowa ścieżka obiegu dokumentu kosztowego z krokami akceptacji, konfiguracją kontrolki Dane tabelaryczne i wygenerowaniem dokumentu do Comarch ERPXL
- **Punkt konfiguracyjny ACD:** Wzorcowa konfiguracja, która jest odpowiedzialna za pobranie ID dokumentu OCR, Załącznika oraz wszystkich kluczowych danych

nagłówkowych (NIP, Kontrahent, Dane kontrahenta, Data zakupu, Data wystawienia, Termin płatności, Forma płatności, Waluta) oraz inicjację dokumentu obiegowego.

[Kliknij tutaj, aby pobrać przykład](#)


Zasada działania

Mechanizm opiera się na **pobieraniu danych z bazy Comarch BPM, które zostały zapisane po przetworzeniu ich przez Comarch OCR w punkcie ACD.**

- Konfiguracja wykorzystuje **ID dokumentu OCR** (zapisane w dedykowanej kontrolce typu **Liczba całkowita**) jako identyfikator do pobrania danych. Kontrolka tabelaryczna **obserwuje to pole**, reagując na jego wypełnienie.
- Kluczowym elementem jest **zapytanie SQL** umieszczone na zakładce **Inicjowanie Wartości** kontrolki tabelarycznej. Zapytanie jest filtrowane za pomocą parametru **@ID_OCR@**, dzięki czemu pobiera ono wiersze z właściwego dokumentu OCR z tabeli **do.OC_Document_Items** w bazie BPM.
- Na podstawie zwróconych wierszy z tabeli **do.OC_Document_Items**, kontrolka **Dane tabelaryczne** jest automatycznie wypełniana pozycjami faktury (Nazwa, Cena, Ilość, Jednostka miary, Netto, Stawka VAT, Brutto, Kwota VAT).
- W zapytaniu SQL zawarta jest niezbędna **logika CASE WHEN**, która konwertuje wartości tekstowe (np. Stawki VAT oraz Jednostki miary) na identyfikatory liczbowe, których BPM wymaga do poprawnego działania mechanizmów systemowych i wyświetlania danych.


Konfiguracja przykładowa

1. **Utworzono kontrolkę ID_OCR** : Na karcie obiegu dodano kontrolkę typu **Liczba całkowita** o nazwie (identyfikatorze) **ID_OCR**, do której będzie zapisywane ID OCR-owanego dokumentu.

Techniczne 

[Liczba całkowita] ID_OCR

Liczba całkowita

 Kontrolkę można uzupełnić liczbą z zakresu od -2 147 483 648 do 2 147 483 647

Nazwa wyświetlana	<input type="text" value="ID_OCR"/>
Nazwa (identyfikator)	<input type="text" value="ID_OCR"/>
Prezentuj na całej szerokości	<input type="checkbox"/>
Ustaw kontrolkę na początku wiersza	<input checked="" type="checkbox"/>
Pokaż na liście	<input type="checkbox"/>

Kontrolka Liczba całkowita ID_OCR

2. **Utworzono kontrolkę Tabelaryczną**: Dodano kontrolkę typu **Dane tabelaryczne** o nazwie (identyfikatorze) **Elementy** i skonfigurowano w niej następujące kolumny:

- **"Nazwa"** (identyfikator kolumny: **Nazwa**) – typ **Tekst**
- **"Cena"** (identyfikator kolumny: **Cena**) – typ **Liczba stałoprzecinkowa**
- **"Ilość"** (identyfikator kolumny: **Ilosc**) – typ **Liczba stałoprzecinkowa**
- **"Jednostka miary"** (identyfikator kolumny: **JM**) – typ

Lista

- "Netto" (identyfikator kolumny: Netto) – typ Liczba stałoprzecinkowa
- "Stawka VAT" (identyfikator kolumny: StawkaVAT) – typ Lista
- "Brutto" (identyfikator kolumny: Brutto) – typ Liczba stałoprzecinkowa
- "Kwota VAT" (identyfikator kolumny: KwotaVAT) – typ Liczba stałoprzecinkowa

Pozycje dokumentu								
Elementy								
LP	Nazwa	Cena	Ilość	Jednostka miary	Netto	Stawka VAT	Brutto	Kwota VAT
+	X							

Kolumny w kontrolce Dane tabelaryczne

3. W kontrolce typu **Dane tabelaryczne** na zakładce **Listy** zdefiniowano zapytanie pobierające:

- **Stawki VAT:**

```
select Naz_gidLp as Id,
CASE
WHEN Naz_Nazwa='A 23.00' THEN '23%'
WHEN Naz_Nazwa='B 8.00' THEN '8%'
WHEN Naz_Nazwa='C 0.00' THEN '0%'
WHEN Naz_Nazwa='D 0.00' THEN 'ZW'
WHEN Naz_Nazwa='E 0.00' THEN 'NP'
WHEN Naz_Nazwa='F 7.00' THEN '7%'
WHEN Naz_Nazwa='G 5.00' THEN '5%'
END
from CDN.Nazwy
where Naz_GIDTyp = 624 AND trim(substring(Naz_Nazwa,9,2))= ''
```

[Dane tabelaryczne] Elementy

Dane tabelaryczne

OGÓLNE

LISTY

INICJOWANIE WARTOŚCI

OGRANICZENIE EDYCJI

Lista 1



Nazwa

Stawka VAT

Inicjowanie

SQL ERP



Rodzaj identyfikatora

Int



[Pokaż](#) [Usuń](#)

Tryb pracy

Lista



Lista Stawka VAT w kontrolce Dane tabelaryczne


▪ Jednostki miary:

```
SELECT Naz_GIDLp,Naz_Nazwa from cdn.Nazwy where Naz_GIDTyp=144  
;
```

[Dane tabelaryczne] Elementy

Dane tabelaryczne

OGÓLNE LISTY INICJOWANIE WARTOŚCI OGRANICZENIE EDYCJI

Lista 1 

Nazwa JM

Inicjowanie Jednostki miary ▼

Rodzaj identyfikatora Int ▼

[Pokaż](#) [Usuń](#)

Tryb pracy Lista ▼

[Dodaj](#)

Lista Jednostka miary w kontrolce Dane tabelaryczne

4. Dodano obserwację kontrolki: W kontrolce typu **Dane tabelaryczne**, na zakładce **Inicjowanie wartości**, dodano powiązanie do kontrolki **ID_OCR**.

[Dane tabelaryczne] Elementy

[Etykieta] Podsumowanie

Dane tabelaryczne

OGÓLNE

LISTY

INICJOWANIE WARTOŚCI

OGRANICZENIE EDYCJI

Inicjowanie wartości

Inicjowanie

Brak

Kontrolki powiązane

ID_OCR (ID_OCR)



Dodaj

Wzór na wartość

SQL ERP

Ustaw

Powiązanie do kontrolki ID_OCR

5.Zapytanie SQL: W kontrolce typu **Dane tabelaryczne**, na zakładce **Inicjowanie wartości** w sekcji **Wzór na wartość SQL OD** dodano zapytanie, które pobiera dane z tabel **do.OC_Document_Items** oraz **do.OC_Documents** na podstawie przekazanego parametru **@ID_OCR@** (ID dokumentu OCR). W zapytaniu SQL konieczne jest użycie logiki CASE WHEN do konwersji wartości tekstowych **jednostek miary** ('szt') i **stawek VAT** (np. '23.00') na odpowiednie identyfikatory liczbowe akceptowane przez Comarch BPM.

```
If @^SenderControlName@ = 'ID_OCR'
```

```
Begin
```

```
Select
ODI_Name, /*Nazwa*/
ODI_NetUnitPrice, /*Cena netto*/
ODI_Count, /*Ilość*/
CASE
WHEN ODI_Unit='szt.' THEN 1
WHEN ODI_Unit='l' THEN 2
WHEN ODI_Unit='kg' THEN 3
Else 0
END, /*Jednostka miary*/
ODI_NetValue, /* Wartość Netto*/
CASE
WHEN ODI_VatRate='23.00' THEN 1
WHEN ODI_VatRate='8.00' THEN 2
WHEN ODI_VatRate='0.00' THEN 3
WHEN ODI_VatRate='5.00' THEN 7
Else 0
END,
ODI_GrossValue, /*Wartość Brutto*/
[ODI_GrossValue]-[ODI_NetValue], -- kwota VAT
1
```

```
from do.OC_Document_Items I
```

```
join do.OC_Documents D on D.OCD_Id=I.ODI_DocumentId
```

```
where ODI_DocumentId=@ID_OCR@
```

```
END
```

Wyrażenie kontrolki

Uwaga! Zmiana nazwy kontrolki może spowodować nieprawidłowe działanie wyrażenia.

```
1 If @SenderControlName@ = 'ID_OCR'
2 Begin
3 Select
4 ODI_Name, /*Nazwa*/
5 ODI_NetUnitPrice, /*Cena netto*/
6 ODI_Count, /*Ilość*/
7 CASE
8 WHEN ODI_Unit='szt.' THEN 1
9 WHEN ODI_Unit='l' THEN 2
10 WHEN ODI_Unit='kg' THEN 3
11 Else 0
12 END, /*Jednostka miary*/
13
14 ODI_NetValue, /*Wartość Netto*/
15 CASE
16 WHEN ODI_VatRate='23.00' THEN 1
17 WHEN ODI_VatRate='8.00' THEN 2
18 WHEN ODI_VatRate='0.00' THEN 3
19 WHEN ODI_VatRate='5.00' THEN 7
20 Else 0
21 END,
22 ODI_GrossValue, /*Wartość Brutto*/
23 [(ODI_GrossValue)-(ODI_NetValue)], -- kwota VAT
24
25 1
26 from do.OC_Document_Items I
27 join do.OC_Documents D on D.OCD_Id=I.ODI_DocumentId
28 where ODI_DocumentId=@ID_OCR@
29 end
```

Kontrolki

Szukaj

ZAKŁADKA 1

Pozycje dokumentu

[Dane tabelaryczne] Elementy

Dane tabelaryczne

OGÓLNE LISTY INICJOWANIE WARTOŚCI OGRANICZENIE EDYCJI

Inicjowanie wartości

Inicjowanie

Kontrolki powiązane ID_OCR (ID_OCR)

[Dodaj](#)

Wzór na wartość

[Ustaw](#)

Zapisz Zamknij

Zapytanie SQL w kontrolce Dane tabelaryczne na zakładce Inicjowanie wartości

Uwaga

Należy upewnić się, że identyfikatory liczbowe (ID) **stawek VAT** i **jednostek miary** użyte w zapytaniu SQL są zgodne z tymi, które aktualnie obowiązują w bazie Comarch ERP XL współpracującej z Comarch BPM.

6.Punkt ACD: W punkcie konfiguracyjnym ACD (Współpraca z Comarch OCR), wykorzystywanym do inicjowania obiegu, na

zakładce **Kontrolki** dodano kontrolkę **ID_OCR** i przypisano jej odpowiednie zapytanie:

```
select @Id1@
```

Punkt konfiguracyjny

Tryb współpracy Ogólne Lista **Kontrolki** Uprawnienia

Skan dokumentu (Skan)

Numer dokumentu (Numerdokumentu)

Kontrahent (Kontrahent)

Dane kontrahenta (Danekontrahenta)

Waluta (Waluta)

Termin płatności (Terminpłatności)

Forma płatności (Formapłatności)

Data wystawienia (Datawystawienia)

Data zakupu (Datazakupu)

IDOCR (ID_OCR)

Dodaj

```
1 select @Id1@
```

Test zapytania SQL

Składnia:
SELECT int
int - liczba całkowita

Przykład:
BEGIN TRY
select value from tabela

Zapisz Zamknij

Kontrolka ID_OCR w punkcie ACD

Uzupełniona Kontrolka Dane Tabełaryczne

Po pomyślnym zainicjowaniu dokumentu z punktu **ACD**, kontrolka typu **Dane Tabełaryczne** została automatycznie wypełniona. Dla każdej pozycji dokumentu (faktury) odczytanej przez Comarch OCR i zapisanej w tabeli **do.OC_Document_Items**, na formularzu BPM pojawił się **oddzielny, kompletny wiersz** z uzupełnionymi

danymi.

Pozycje dokumentu

Elementy

LP	Nazwa	Cena	Ilość	Jednostka miary	Netto	Stawka VAT	Brutto	Kwota VAT
1	TOWAR1	60,00	4,00	szt.	240,00	23%	295,20	55,20
2	TOWAR2	30,00	7,00	szt.	210,00	23%	258,30	48,30
3	TOWAR3	80,00	7,00	szt.	560,00	23%	688,80	128,80
4	TOWAR4	90,00	8,00	kg	720,00	8%	777,60	57,60

Podsumowanie

Wartość Netto: 1 730,00 Wartość Brutto: 2 019,90

Techniczne

ID_OCR: 5

Kontrolka Dane tabelaryczne uzupełniona pozycjami odczytanym przez Comarch OCR

Rozpoczynasz pracę z Comarch BPM (dawniej DMS) i chcesz dowiedzieć się, jak korzystać z programu? A może masz już podstawową wiedzę o Comarch BPM (dawniej DMS) i chcesz dowiedzieć się więcej?

[Sprawdź Szkolenia Comarch BPM!](#)

[Powrót do początku artykułu](#)

Czy wiesz, że za pomocą procedury SQL do.OnAfterDocumentCreated możliwe jest automatyczne podpięcie wszystkich załączników z e-maila do dokumentu BPM, w tym logiczne rozdzielenie plików zOCR-owanych od pozostałych, które nie podlegają przetwarzaniu?

Zastosowanie

Dzięki tej konfiguracji złożonej z dwóch punktów ACD (Współpraca ze skrzynką pocztową (pobieranie załączników i Współpraca z Comarch OCR) oraz procedury SQL (do.OnAfterDocumentCreated), możliwa jest precyzyjna kontrola nad załącznikami pobranymi z maila, które trafiają do tworzonego dokumentu BPM. Mechanizm ten ma zastosowanie w procesach, gdzie dokumenty są pobierane ze skrzynki pocztowej i w mailu oprócz dokumentu PDF, który ma zostać zOCR-owany, są również inne załączniki, które powinny zostać podpięte podczas tworzenia dokumentu BPM, ale nie są przeznaczone do

przetwarzania przez OCR.

Celem jest pobranie **wszystkich załączników** dołączonych do maila i podpięcie ich na dokument BPM z zachowaniem logicznej separacji oraz zapewnieniem, że plik główny jest podpięty pod kontrolkę o nazwie **Załącznik**, a pozostałe pliki z maila są podpięte pod osobną kontrolkę o nazwie **Pozostałe załączniki**.

Zawartość przykładu

- **Definicja Typu Obiegu (Workflow):** Wzorcowa ścieżka obiegu dokumentu kosztowego z kontrolką typu **Załącznik** o nazwie **Załącznik**, do której będzie pobierany plik po przetworzeniu przez **OCR** oraz drugą kontrolką tego samego typu o nazwie **Pozostałe załączniki**, do której będą pobierane pozostałe pliki z maila.
- **Punkt konfiguracyjny ACD:** Wzorcowa konfiguracja, która jest odpowiedzialna za pobranie ID dokumentu OCR, Załącznika oraz wszystkich kluczowych danych nagłówkowych (NIP, Kontrahent, Dane kontrahenta, Data zakupu, Data wystawienia, Termin płatności, Forma płatności, Waluta) oraz inicjację dokumentu obiegowego.
- **Procedura SQL (do.OnAfterDocumentCreated):** Procedura odpowiedzialna za logiczną segregację, eliminację wtórnych wystąpień oraz wstawienie załączników do odpowiednich kontrolkek na karcie obiegu.

Uwaga

- W **definicji typu obiegu** w kontrolkach są **zapytania SQL**, które odwołują się do bazy danych **Comarch ERP XL**

- W punkcie ACD dla zapytania pobierającego **Kontrahenta** należy wskazać właściwą nazwę bazy **ERPXL**
- Punkt **ACD Współpraca ze skrzynką pocztową (pobieranie załączników)** należy skonfigurować we **własnym zakresie**.

[Kliknij tutaj, aby pobrać przykład](#)

Zasada działania

Mechanizm opiera się na sekwencyjnym działaniu **dwóch punktów ACD i procedury SQL**:

- **Etap I (ACD Współpraca ze skrzynką pocztową)**: Na skrzynkę zostaje wysłany mail, który zawiera **główny plik pdf** oraz **pozostałe pliki**. Po pobraniu wiadomości użytkownik **pozostawia** tylko plik pdf, który ma zostać przetworzony przez OCR, pozostałe pliki, które nie będą przetwarzane przez OCR **usuwa**.
- **Etap II (ACD Współpraca z Comarch OCR)**: Plik główny PDF po pobraniu z maila zostaje zaczytany przez Comarch OCR. Następnie wygenerowany do Comarch BPM.
- **Etap III (Procedura SQL – OnAfterDocumentCreated)**: Podczas generowania dokumentu w BPM z punktu **ACD Współpraca z Comarch OCR** jest wywoływana procedura **do.OnAfterDocumentCreated**, która podpiną przetworzony przez **OCR** plik PDF pod kontrolkę **Załącznik**, a do kontrolki **Pozostałe załączniki** są podpinane pozostałe pliki z warunkiem wykluczającym plik główny.

Konfiguracja przykładowa

1. **Utworzono kontrolkę Załącznik:** Na karcie obiegu dodano kontrolkę typu **Załącznik** o nazwie **Załącznik** (identyfikatorze **Zalacznik**), do której będzie podpinany przetworzony przez OCR dokument.

[Załącznik] Załącznik

Załącznik

OGÓLNE ARCHIWA

Nazwa wyświetlana: Załącznik

Nazwa (identyfikator): Zalacznik

Podpisy elektroniczne PDF:

Mobilny podpis:

Zablokuj usuwanie załączników:

Prezentuj na całej szerokości:

Ustaw kontrolkę na początku wiersza:

Wersjonowanie pliku: Nie

Kontrolka typu Załącznika o nazwie Załącznik

2. **Utworzono kontrolkę Załącznik:** Na karcie obiegu dodano kontrolkę typu **Załącznik** o nazwie **Pozostałe załączniki** (identyfikatorze **PozostaleZalaczniki**), do której będą

podpinane pozostałe pliki z maila.

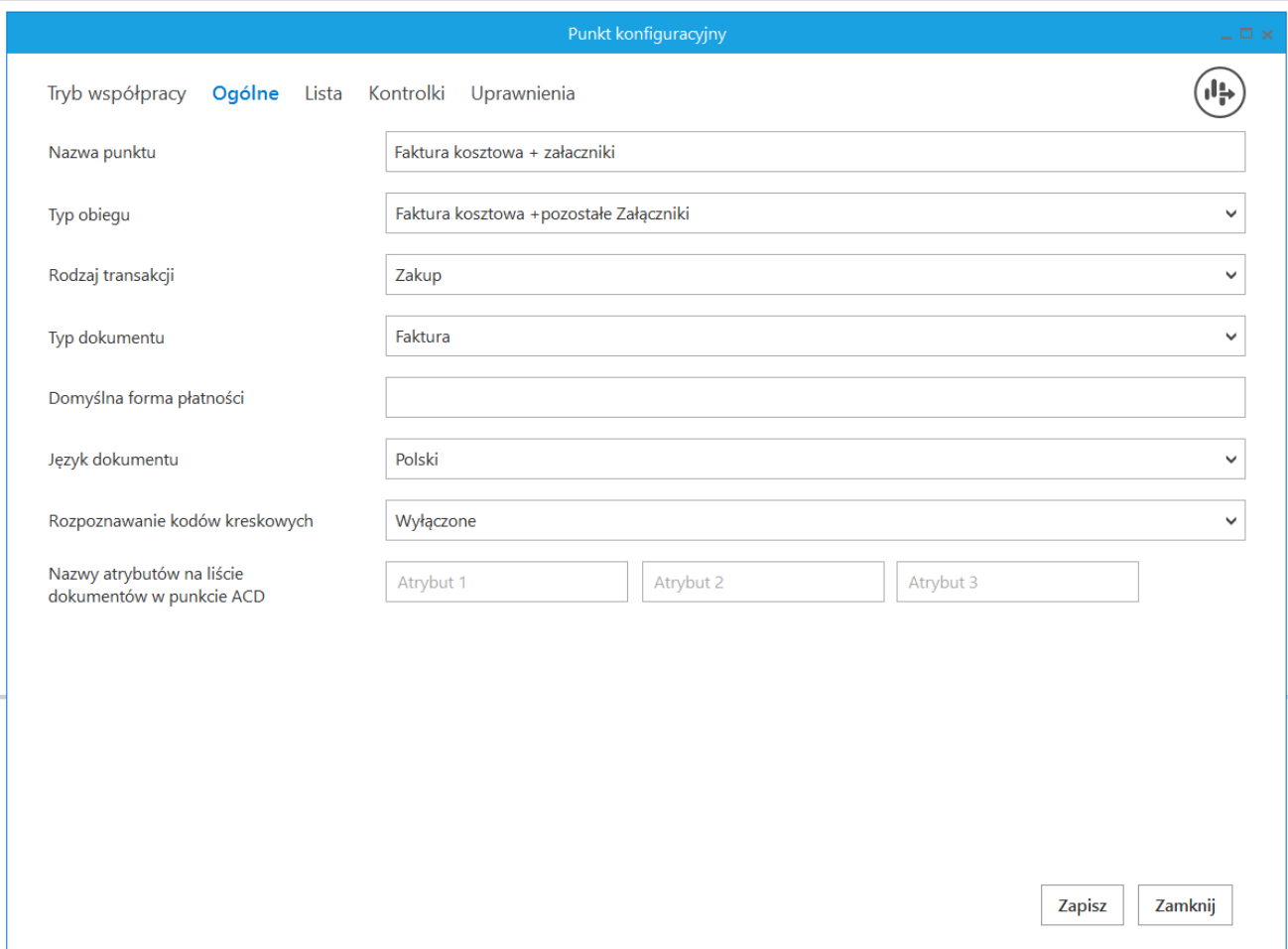
The image shows a screenshot of a software interface for configuring an attachment type. At the top, a grey header bar contains the text "[Załącznik] Pozostałe załączniki". Below this is a blue horizontal bar with the word "Załącznik" in white. The main content area has two tabs: "OGÓLNE" (selected) and "ARCHIWA". Under the "OGÓLNE" tab, there are several configuration options:

Nazwa wyświetlana	<input type="text" value="Pozostałe załączniki"/>
Nazwa (identyfikator)	<input type="text" value="PozostałeZalaczniki"/>
Podpisy elektroniczne PDF	<input type="checkbox"/>
Mobilny podpis	<input type="checkbox"/>
Zablokuj usuwanie załączników	<input type="checkbox"/>
Prezentuj na całej szerokości	<input type="checkbox"/>
Ustaw kontrolkę na początku wiersza	<input type="checkbox"/>
Wersjonowanie pliku	<input type="text" value="Nie"/>

Kontrolka typu Załącznika o nazwie Pozostałe załączniki

3. **Utworzono kontrolkę ID_dokumentuOCR:** Na karcie obiegu dodano kontrolkę typu **Liczba całkowita** o nazwie (identyfikatorze) **ID_dokumentuOCR**, do której będzie zapisywane ID OCR-owanego dokumentu.

4. **Punkt ACD:** Utworzono punkt: **ACD – Współpraca z Comarch OCR** do odczytywania danych z dokumentów pdf.



The screenshot shows a configuration window titled 'Punkt konfiguracyjny'. At the top, there are navigation tabs: 'Tryb współpracy', 'Ogólne' (selected), 'Lista', 'Kontrolki', and 'Uprawnienia'. Below the tabs, there are several configuration fields:

- Nazwa punktu: Faktura kosztowa + załączniki
- Typ obiegu: Faktura kosztowa + pozostałe Załączniki
- Rodzaj transakcji: Zakup
- Typ dokumentu: Faktura
- Domyślna forma płatności: (empty)
- Język dokumentu: Polski
- Rozpoznawanie kodów kreskowych: Wyłączone
- Nazwy atrybutów na liście dokumentów w punkcie ACD: Atrybut 1, Atrybut 2, Atrybut 3

At the bottom right, there are two buttons: 'Zapisz' and 'Zamknij'.

Punkt ACD Współpraca z Comarch OCR – zakładka Ogólne

5. **Utworzono kontrolki:** W punkcie ACD na zakładce **Kontrolki** dodano kontrolki:

- **Załącznik (identyfikator Załącznik)**, do której będzie podpinany przetworzony przez OCR dokument i przypisano zapytanie:

```
select OCD_SourceContent, OCD_FileName from D0.OC_Documents  
left join do.DF_WorkAPProcess on OCD_ID = APW_DocumentID1  
where OCD_ID=@ID1@
```

- **ID_dokumentuOCR**, do której będzie pobierane ID dokumentu OCR i przypisano zapytanie:

```
select @Id1@
```

Punkt konfiguracyjny

Tryb współpracy Ogólne Lista **Kontrolki** Uprawnienia

Załącznik (Zalacznik) Pobierz plik z bazy danych

```
1:
2:
3: select OCD_SourceContent, OCD_FileName from
4: DO_OC_Documents left join do.DF_WorkAPProcess on
5: OCD_ID = APW_DocumentID1
6: where OCD_ID=@ID1@
```

Test zapytania SQL

Składnia:
SELECT Varbinary, varchar(100)
Varbinary – obraz skanu zapisany w bazie
varchar(100) - typ pliku
Przykład:
BEGIN TRY
select scan, 'pdf' from tabela
where Id = @Id1@
END TRY
BEGIN CATCH

Zapisz Zamknij

Punkt ACD Współpraca z Comarch OCR – zakładka Kontrolki

6. **Punkt ACD:** Utworzono punkt **Współpraca ze skrzynką pocztową (pobieranie załączników)**.

Punkt konfiguracyjny

Tryb współpracy Ogólne Lista Uprawnienia

Tryb współpracy: Współpraca ze skrzynką pocztową (pobieranie załączników)

Typ konta: IMAP

Serwer poczty przychodzącej: [input type="text"] Port: 993

Metoda uwierzytelniania: Login i hasło Metoda szyfrowania: TLS 1.2

Login: [input type="text"]

Hasło: [input type="password"] Pokaż hasło

Zapisz Zamknij

Punkt ACD – Współpraca ze skrzynką pocztową (pobieranie załączników)

7. Procedura SQL (do.OnAfterDocumentCreated): Na bazie **BPM** w **Management Studio** w procedurze **do.OnAfterDocumentCreated** został uruchomiony skrypt, który do kontrolki **Załącznik** podcina **zOCR-owany dokument**, a pod kontrolkę **Pozostałe załączniki** podcina pozostałe pliki z maila.

-----Pobieranie załączników z maila-----

```
DECLARE @DWCID_Glowne INT
```

```
DECLARE @DWCID_Inne INT
```

```
DECLARE @DDSID INT
```

```
DECLARE @OCDID INT
```

```
DECLARE @MLM_Id INT
```

```
DECLARE @MLA_Id_Glownego INT
```

```
DECLARE @OCD_FileName_String VARCHAR(MAX)
```

```
/* 1. POBIERANIE ZMIENNYCH */
```

```
/* Pobranie ID Kontrolki 'Załączniki' (kontrolka główna) */
```

```
SELECT @DWCID_Glowne = DWC_ID FROM do.R0_ControlsValue WHERE  
DWD_ID = @DocumentId AND DKO_Name = 'Załącznik'
```

```
/* Pobranie ID Kontrolki 'Pozostałe załączniki' (kontrolka  
pomocnicza) */
```

```
SELECT @DWCID_Inne = DWC_ID FROM do.R0_ControlsValue WHERE  
DWD_ID = @DocumentId AND DKO_Name = 'PozostałeZałączniki'
```

```
/*Pobranie ID dokumentu przetworzonego przez OCR(klucz do  
identyfikacji pliku głównego)*/
```

```
SELECT @OCDID = Value FROM do.R0_ControlsValue WHERE DWD_ID =  
@DocumentId AND DKO_Name = 'ID_dokumentuOCR'
```

```
/* Pobranie ID etapu (Stage ID), na którym znajduje się  
obecnie dokument */
```

```
SELECT @DDSID = DDS_ID FROM do.R0_StageActive WHERE DWD_ID =  
@DocumentId
```

```
/* 2. POBRANIE ID GŁÓWNEGO ZAŁĄCZNIKA (@MLA_Id_Glownego) -  
Poprawne Parsowanie */
```

```

SELECT @OCD_FileName_String = OCD_FileName

FROM do.OC_Documents

WHERE OCD_id = @OCDID

/* Użycie parsowania (Trzeci segment to MLA_Id, pierwszy to
MLM_Id) */

DECLARE @Pos1 INT = CHARINDEX('_', @OCD_FileName_String);

DECLARE @Pos2 INT = CHARINDEX('_', @OCD_FileName_String, @Pos1
+ 1);

DECLARE @Pos3 INT = CHARINDEX('_', @OCD_FileName_String, @Pos2
+ 1);

IF @Pos3 > 0

BEGIN

SET @MLM_Id = CAST(SUBSTRING(@OCD_FileName_String, 0, @Pos1)
AS INT);

SET @MLA_Id_Głownego = CAST(

SUBSTRING(

@OCD_FileName_String,

@Pos2 + 1,

@Pos3 - @Pos2 - 1

)

AS INT);

END

ELSE

```

```
BEGIN
```

```
SET @MLM_Id = 0;
```

```
SET @MLA_Id_Głownego = 0;
```

```
END
```

```
/* 3. CZYSZCZENIE: USUŃ PLIKI WCZEŚNIEJ WSTAWIONE DO KONTROLKI  
GŁÓWNEJ
```

```
To jest konieczne, aby usunąć plik o "długiej nazwie"  
wstawiony przez system OCR */
```

```
DELETE FROM do.DF_WorkCFBin
```

```
WHERE DWB_DWCID = @DWCID_Głowne;
```

```
/* 4. CZĘŚĆ 1: Wstawianie GŁÓWNEGO ZAŁĄCZNIKA (Załączniki)
```

```
Cel: Wstaw oryginalny plik (krótsza nazwa) */
```

```
INSERT INTO do.DF_WorkCFBin (DWB_ID, [DWB_DWCID], [DWB_BIN],  
[DWB_Name], [DWB_Size], [DWB_Archival], [DWB_CreateDate],  
[DWB_CompanyId], [DWB_ModificationData], [DWB_DDSID],  
[DWB_DCDID], [DWB_AngleOfRotation])
```

```
SELECT
```

```
    NEWID(),
```

```
    @DWCID_Głowne, /* Podpinanie pod główną kontrolkę */
```

```
    MLA_Content,
```

```
    MLA_FileName, /* Wstawia KRÓTSZĄ, ORYGINALNĄ NAZWĘ */
```

```
    DATALENGTH(MLA_Content),
```

```
    0,
```

```

    GETDATE(),

    1,

    GETDATE(),

    @DDSID,

    1,

    0

FROM do.ML_Mail_Attachments

WHERE

MLA_MailId = @MLM_Id

/* WARUNEK UNIKALNY: TYLKO ten załącznik, który ma ID głównego
dokumentu */

AND MLA_Id = @MLA_Id_Głownego

/* 5. CZĘŚĆ 2: Wstawianie POZOSTAŁYCH ZAŁĄCZNIKÓW
(PozostałeZałączniki)

-- Cel: Wstaw WSZYSTKO inne, co NIE jest plikiem GŁÓWNYM. */

INSERT INTO do.DF_WorkCFBin (DWB_ID, [DWB_DWCID], [DWB_BIN],
[DWB_Name], [DWB_Size], [DWB_Archival], [DWB_CreateDate],
[DWB_CompanyId], [DWB_ModificationData], [DWB_DDSID],
[DWB_DCDID], [DWB_AngleOfRotation])

SELECT

    NEWID(),

    @DWCID_Inne, -- Podpinanie pod dodatkową kontrolkę

    MLA_Content,

```

```

    MLA_FileName,
    DATALENGTH(MLA_Content),
    0,
    GETDATE(),
    1,
    GETDATE(),
    @DDSID,
    1,
    0
FROM do.ML_Mail_Attachments
WHERE
    MLA_MailId = @MLM_Id

    /* WARUNEK WYKLUCZAJĄCY GŁÓWNY DOKUMENT: Wyklucz plik,
    który ma to samo MLA_Id */

    AND MLA_Id <> @MLA_Id_Głownego

    /* WYKLUCZENIA (Anty-Smime) */

    AND MLA_FileName NOT LIKE '%.smime'

    AND MLA_FileName NOT LIKE '%smime%'

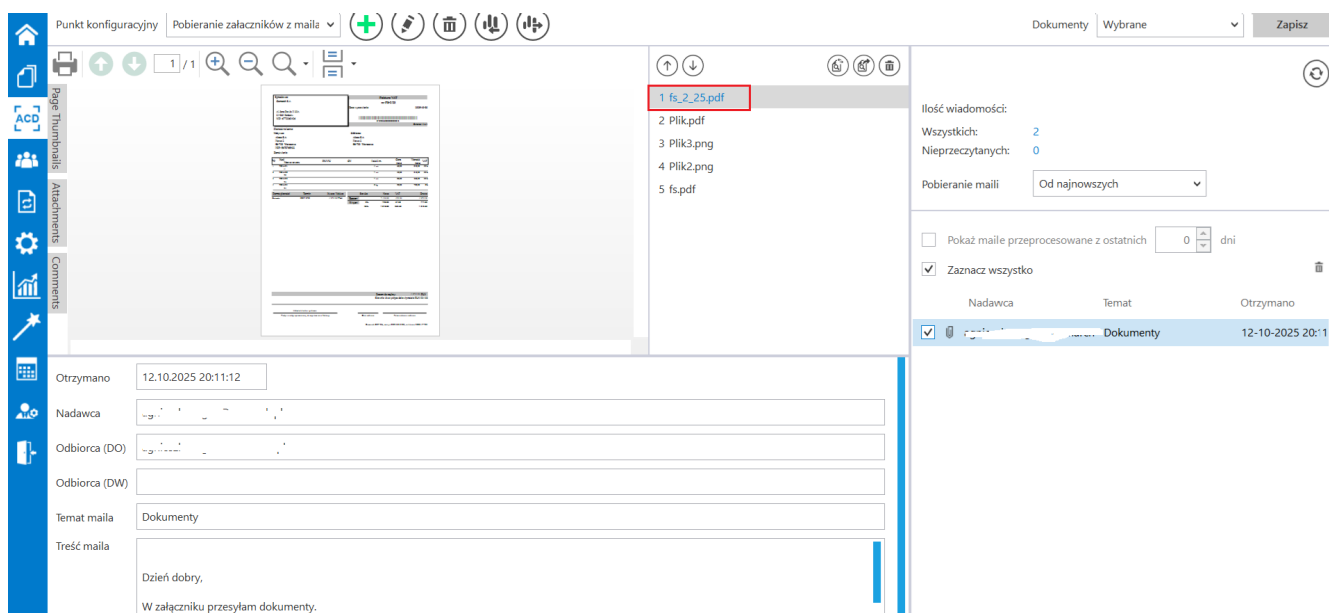
    AND MLA_FileName NOT LIKE '%.p7s'

    AND MLA_FileName NOT LIKE 'ATT%DOC'

```

Podpięcie dokumentów pod odpowiednie kontrolki typu Załącznik po utworzeniu dokumentu z punktu ACD Comarch OCR

1. Do punktu ACD **Współpraca ze skrzynką pocztową (pobieranie załączników)** pobrano wiadomość e-mail z załącznikami. Tylko jeden plik ma zostać przetworzony przez OCR, a pozostałe powinny zostać usunięte z listy ręcznie przez użytkownika.



Punkt ACD – pobranie wiadomości e-mail z załącznikami

2. Po usunięciu pozostałych plików, które nie podlegają przetworzeniu przez **OCR**, został zaznaczony mail i dokument z listy zapisano do folderu, który jest wskazany z powiązaniem punktem **ACD Współpraca z Comarch OCR (Faktura kosztowa +**

załączniki).

Punkt konfiguracyjny Pobieranie załączników z maila

Dokumenty Wybrane Zapisz

1 fs_2_25.pdf

Ilość wiadomości:
Wszystkich: 29
Nieprzeczytanych: 28

Pobieranie maili: Od najnowszych

Pokaż maile przetworzone z ostatnich 0 dni
 Zaznacz wszystko

Nadawca	Temat	Otrzymano
<input checked="" type="checkbox"/>	Dokumenty	12-10-2025 20:11

Otrzymano: 12.10.2025 20:11:12

Nadawca: ...

Odbiorca (DO): ...

Odbiorca (DW): ...

Temat maila: Dokumenty

Treść maila: Dzień dobry,
W załączniku przesyłam dokumenty.

Zapis dokumentu pdf do przetworzenia

3. Dokument został pobrany przez punkt **OCR (Faktura kosztowa + załączniki)**. Po zaznaczeniu dokumentu i kliknięciu na przycisk **Generuj** dokument został utworzony w **Comarch BPM** w ramach typu obiegu wskazanego w tym punkcie.

Punkt konfiguracyjny Faktura kosztowa + załączniki

Dokumenty Wybrane Generuj

1/1

Włącz procesowanie dokumentów
Przepracowane: 1

Zaznacz wszystko Szukaj

Dokument

FS-2/25

Data utworzenia : 13-10-2025

Numer dokumentu (Numer_dokumentu) FS-2/25
 Kontrahent (Kontrahent) [COMARCH] COMARCH SPÓŁKA AKCYJNA
 Dane kontrahenta (Dane_kontrahenta) Aleja Jana Pawła II 39 A 31-864 Kraków

Waluta (Waluta) PLN
 Termin płatności (Termin_platnosci) 2025-10-03

Forma płatności (Forma_platnosci) Gotówka
 Data wystawienia (Data_wystawienia) 2025-10-03
 Data zakupu (Data_zakupu) 2025-10-03

ID_dokumentuOCR (ID_dokumentuOCR) 7

Generowanie dokumentu

Dokumenty Szukaj

NUMER	NAZWA OBIEGU	NAZWA ETAPU	DATA PRZEKAZANIA	STATUS	ETAP
FA+ACD/3/10/2025	Faktura kosztowa + pozostałe Załączniki	Rejestracja dokumentu	2025-10-13	Niezakończony	Przypisany do Ciebie

Wszystkie (1)
 PROCEDURE (1)
 do.OnAfterDocumentCreated ('
 Faktura kosztowa + pozostz

Utworzony dokument na liście dokumentów

4. Dokument główny został podpięty pod kontrolkę **Załącznik**, a pozostałe pliki z maila pod kontrolkę **Pozostałe załączniki**.

FA+ACD/6/10/2025

← 🗑️ ↻ 🖨️

Dokumenty

Data utworzenia : 13-10-2025

Dane ogólne

Załącznik

PDF fs_2_25

+

Pozostałe załączniki

PDF Plik

PDF fs

Plik2

Plik3

+

Numer dokumentu

FS-2/25

Data wystawienia

3.10.2025

Data zakupu

3.10.2025

Data wpływu

13.10.2025

Podpięcie dokumentu przetworzonego przez OCR wraz z pozostałymi załącznikami.

Rozpoczynasz pracę z Comarch BPM (dawniej DMS) i chcesz dowiedzieć się, jak korzystać z programu? A może masz już podstawową wiedzę o Comarch BPM (dawniej DMS) i chcesz dowiedzieć się więcej?

[Sprawdź Szkolenia Comarch BPM!](#)

[Powrót do początku artykułu](#)

Czy wiesz, że po dodaniu wiersza w kontrolce typu Dane tabelaryczne poszczególne kolumny mogą zainicjować się wartością domyślną?

Zastosowanie

Dzięki tej konfiguracji, proces **wprowadzania danych** jest znacząco **przyspieszony** w przypadku dokumentów wymagających wielokrotnego dodawania tych samych lub podobnych pozycji. Głównym celem jest **automatyczne** wypełnienie kolumn **nowo dodanego wiersza** domyślnymi wartościami (np. nazwa, ilość, stawka VAT). **Eliminuje** to błędy i powtarzalne czynności użytkownika, co pozwala pracownikom skupić się na **weryfikacji danych**, a nie na ich rutynowym wprowadzaniu. Dodatkową, unikalną korzyścią tego przykładu jest to, że zawiera on w sobie pełną logikę **dynamicznego i dwukierunkowego przeliczania kwot (Netto/Brutto)** w wierszu, dzięki dodatkowemu zapytaniu SQL.

Zawartość przykładu

- **Definicja Typu Obiegu (Workflow):** Wzorcowa ścieżka obiegu dokumentu kosztowego z krokami akceptacji i konfiguracją kontrolki Dane tabelaryczne i wygenerowaniem dokumentu do Comarch ERPXL

[Kliknij tutaj, aby pobrać przykład](#)

Zasada działania

Mechanizm opiera się na monitorowaniu **zdarzenia dodania nowego wiersza** i uruchamianiu zapytania SQL, które w odpowiedzi wypełnia pola domyślnymi wartościami.

- Kontrolka **Dane Tabełaryczne** jest skonfigurowana tak, aby **obserwować samą siebie** i reagować na zdarzenie wywołane przez dodanie **nowego wiersza** za pomocą przycisku „+” (**Dodaj**).
- Na zakładce **Inicjowanie wartości** umieszczone jest zapytanie **SQL**, które sprawdza, czy kolumna ma **wartość -1** (@Elementy_Column@ = -1). **Wartość -1** oznacza, że operacja dotyczy **dodania nowego wiersza**.
- Wewnątrz bloku **BEGIN** zapytanie SQL używa komendy **UPDATE** @Elementy@ do ustawienia domyślnych wartości (np. Nazwa = 'a-vista', Ilosc = 1, StawkaVAT=1) dla **nowo dodanej pozycji**.
- Kolumna wynikowa **Netto** została ustawiona w trybie „**Do odczytu**”, aby użytkownik nie mógł ręcznie jej edytować.

Konfiguracja przykładu

1. **Utworzono kontrolkę Tabełaryczną:** Dodano kontrolkę typu **Dane tabelaryczne** o nazwie (identyfikatorze) **Elementy** i skonfigurowano w niej następujące kolumny:

- **"Nazwa"** (identyfikator kolumny: **Nazwa**) – typ **Tekst**
- **"Cena"** (identyfikator kolumny: **Cena**) – typ **Liczba stałoprzecinkowa**
- **"Ilość"** (identyfikator kolumny: **Ilosc**) –typ **Liczba stałoprzecinkowa**
- **"Netto"** (identyfikator kolumny: **Netto**) – typ **Liczba**

stałoprzecinkowa

- "Stawka VAT" (identyfikator kolumny: StawkaVAT) – typ **Lista**
- "Brutto" (identyfikator kolumny: Brutto) – typ **Liczba stałoprzecinkowa**

Pozycje dokumentu

Elementy

LP	Nazwa	Cena	Ilość	Netto	Stawka VAT	Brutto
						

Kolumny w kontrolce Dane tabelaryczne

2. W kontrolce typu **Dane tabelaryczne** na zakładce **Listy** zdefiniowano zapytanie pobierające:

▪ Stawki VAT:

```
select Naz_gidLp as Id,
CASE
  WHEN Naz_Nazwa='A 23.00' THEN '23%'
  WHEN Naz_Nazwa='B 8.00' THEN '8%'
  WHEN Naz_Nazwa='C 0.00' THEN '0%'
  WHEN Naz_Nazwa='D 0.00' THEN 'ZW'
  WHEN Naz_Nazwa='E 0.00' THEN 'NP'
  WHEN Naz_Nazwa='F 7.00' THEN '7%'
  WHEN Naz_Nazwa='G 5.00' THEN '5%'
END
from CDN.Nazwy
where Naz_GIDTyp = 624 AND trim(substring(Naz_Nazwa,9,2))=''
```

[Dane tabelaryczne] Elementy

Dane tabelaryczne

OGÓLNE

LISTY

INICJOWANIE WARTOŚCI

OGRANICZENIE EDYCJI

Lista 1



Nazwa

Stawka VAT

Inicjowanie

SQL ERP



Rodzaj identyfikatora

Int



[Pokaż](#) [Usuń](#)

Tryb pracy

Lista



Lista Stawka VAT w kontrolce Dane tabelaryczne

3. **Dodano obserwację kontrolki:** W kontrolce **Dane tabelaryczne**, na zakładce **Inicjowanie wartości**, dodano powiązanie do samej siebie.

Dane tabelaryczne

OGÓLNE

LISTY

INICJOWANIE WARTOŚCI

OGRANICZENIE EDYCJI

Inicjowanie wartości

Inicjowanie

Brak

Kontrolki powiązane

Elementy (Elementy)



Dodaj

Wzór na wartość

SQL ERP

Ustaw

Powiązanie kontrolki do samej siebie

4. **Dodano zapytanie inicjujące SQL:** W kontrolce typu **Dane tabelaryczne**, na zakładce **Inicjowanie wartości** w sekcji **Inicjowanie SQL ERP** dodano zapytanie, które zawiera warunek sprawdzający dodanie nowego wiersza **IF @Elementy_Column@ = -1** oraz instrukcję **update**, która ustawia wartości domyślne dla kolumn (**Nazwa**, **Ilość**, **Stawka VAT**) w bieżącym wierszu.

Oprócz inicjacji nowego wiersza, zapytanie zawiera również logikę **dwukierunkowego przeliczania wartości** (Netto, Brutto, Cena) dla już istniejących wierszy, co czyni ten wzorzec gotowym rozwiązaniem dla każdej kontrolki z kalkulacjami finansowymi.

```
IF @^SenderControlName@ = 'Elementy'
```

```
IF @Elementy_Column@ = -1 /*[1] LOGIKA: INICJACJA NOWEGO  
WIERSZA (WYZWALANE KLIKNIĘCIEM '+') */
```

```
BEGIN
```

```
/*Ustawienie domyślnych wartości dla nowego wiersza */
```

```
UPDATE @Elementy@
```

```
SET [Nazwa] = 'a-vista',
```

```
[Ilosc] = 1,
```

```
[StawkaVAT] = 1 /*Identyfikator stawki domyślnej (ID=1 -  
stawka 23%)*/*
```

```
WHERE POS = @Elementy_Row@
```

```
/*!!! OBOWIĄZKOWE: Odświeżenie wiersza w kontrolce Dane  
Tabelaryczne !!! */
```

```
SELECT * FROM @Elementy@
```

```
END
```

```
ELSE IF @Elementy_Column@=2 OR @Elementy_Column@=3 OR  
@Elementy_Column@=5
```

```
/*[2] LOGIKA: PRZELICZANIE W PRZÓD (Cena/Ilość/Stawka VAT ->  
Netto/Brutto) */
```

```
BEGIN
```

```
DECLARE @VAT AS DECIMAL(4, 2)
```

```
/*Pobranie stawki VAT z tabel CDN.Nazwy */
```

```
SELECT @VAT = CAST(SUBSTRING(Naz_Nazwa1, 0, CHARINDEX(' ',  
Naz_Nazwa1))
```

```
AS DECIMAL(4, 2)) / 100
```

```
FROM CDN.Nazwy
```

```
WHERE Naz_GIDTyp = 624 AND Naz_GIDLp = (SELECT [StawkaVAT]  
FROM @Elementy@
```

```
WHERE POS = @Elementy_Row@)
```

```
/*Wyliczenie i aktualizacja netto i brutto*/
```

```
UPDATE @Elementy@
```

```
SET [Netto] = (Ilosc * Cena),
```

```
[Brutto] = (Ilosc * Cena) * (@VAT + 1)
```

```
WHERE POS = @Elementy_Row@
```

```
/*!!! OBOWIĄZKOWE: Odświeżenie wiersza w kontrolce Dane  
Tabelaryczne !!! */
```

```
SELECT * FROM @Elementy@
```

```
END
```

```
ELSE BEGIN
```

```
IF @Elementy_Column@=3 OR @Elementy_Column@=5 OR  
@Elementy_Column@=6 -- [3] LOGIKA: PRZELICZANIE W TYŁ (Brutto
```

-> Cena/Netto)

/*Pobranie stawki VAT*/

```
SELECT @VAT = CAST(SUBSTRING(Naz_Nazwa1, 0, CHARINDEX(' ',  
Naz_Nazwa1))
```

```
AS DECIMAL(4, 2)) / 100
```

```
FROM CDN.Nazwy
```

```
WHERE Naz_GIDTyp = 624 AND Naz_GIDLp = (SELECT [StawkaVAT]  
FROM @Elementy@
```

```
WHERE POS = @Elementy_Row@)
```

/*Aktualizacja ceny i netto*/

```
UPDATE @Elementy@
```

```
SET [Cena] = [Brutto] / (@VAT + 1),
```

```
[Netto] = [Ilosc] * ([Brutto] / (@VAT + 1))
```

```
WHERE POS = @Elementy_Row@
```

/*!!! OBOWIĄZKOWE: Odświeżenie wiersza w kontrolce Dane
Tabelaryczne !!!*/

```
SELECT * FROM @Elementy@
```

```
END
```

Wyrażenie kontrolki

Uwaga! Zmiana nazwy kontrolki może spowodować nieprawidłowe działanie wyrażenia.

```

1 IF @SenderControlName@ = 'Elementy'
2 IF @Elementy_Column@ = -1 /*[1] LOGIKA: INICJACJA NOWEGO WIERSZA (WYZNALANE KLIKNIĘCIEM '+') */
3 BEGIN
4 /*Ustawienie domyślnych wartości dla nowego wiersza */
5 UPDATE @Elementy@
6 SET [Nazwa] = 'a-vista',
7 [Ilość] = 1,
8 [StawkaVAT] = 1 /*Identyfikator stawki domyślnej (ID=1 - stawka 23%)/
9 WHERE POS = @Elementy_Row@
10
11 /*!!! OBOWIĄZKOWE: Odświeżenie wiersza w kontrolce Dane Tabełaryczne !!! */
12 SELECT * FROM @Elementy@
13 END
14
15
16
17 ELSE IF @Elementy_Column@=2 OR @Elementy_Column@=3 OR @Elementy_Column@=5
18 /*[2] LOGIKA: PRZELICZANIE W PRZÓD (Cena/Ilość/Stawka VAT -> Netto/Brutto) */
19 BEGIN
20 DECLARE @VAT AS DECIMAL(4, 2)
21
22 /*Pobranie stawki VAT z tabeli CDN.Nazwy */
23 SELECT @VAT = CAST(SUBSTRING(Naz Nazwa1, 0, CHARINDEX(' ', Naz Nazwa1)))

```

Kontrolki

- ^UserId
- ^DocumentId
- ^WorkflowId
- ^StageId
- ^InitSenderControlName
- ^SenderControlName
- Dane_kontrahenta
- Data_wplywu

Składnia:

SELECT

varchar(900) - Nazwa

decimal(22,4) - Cena

decimal(22,4) - Ilość

decimal(22,4) - Netto

int - Id Stanowiska (StawkaVAT)

decimal(22,4) - Brutto

POS - pozycja (opcjonalny numer wiersza)

Przykład użycia:

```

BEGIN TRY
SELECT 'txt1', 345.324, 345.324, 345.324, 23,
245 324

```

KONTROLKI

ZAKŁADKA 1

Pozycje dokumentu

[Dane tabelaryczne] Elementy

Dane tabelaryczne

OGÓLNE LISTY **INICJOWANIE WARTOŚCI** OGRANICZENIE EDYCJI

Inicjowanie wartości

Inicjowanie

Kontrolki powiązane

Elementy (Elementy)

Wzór na wartość

Zapytanie SQL w kontrolce Dane tabelaryczne na zakładce Inicjowanie wartości

5. Tryb pracy: W kontrolce typu Dane tabelaryczne, na zakładce **Ograniczenie edycji** kolumnę „Netto” ustawiono w trybie „Do odczytu”.

Dane tabelaryczne								
OGÓLNE	LISTY	INICJOWANIE WARTOŚCI	OGRANICZENIE EDYCJI					
Zaznaczenie opcji spowoduje zablokowanie funkcjonalności lub edycji poniższych parametrów.								
	<input type="checkbox"/> Zaznacz wszystko	<input type="checkbox"/> Zaznacz wszystko	<input type="checkbox"/> Zaznacz wszystko	<input type="checkbox"/> Zaznacz wszystko	<input type="checkbox"/> Zaznacz wszystko	<input checked="" type="checkbox"/> Zaznacz wszystko	<input type="checkbox"/> Zaznacz wszystko	<input type="checkbox"/> Zaznacz wszystko
Etap	Zablokuj dodanie wierszy	Zablokuj usuwanie wierszy	Nazwa Nazwa	Cena Cena	Ilość Ilość	Netto Netto	StawkaVAT Stawka VAT	Brutto Brutto
Rejestracja dokumentu	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Dekreatacja kosztów	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Akceptacja kierownika	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Księgowość	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Odrzucono	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Archiwum	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Kolumna Netto w trybie „Do odczytu”

Uzupełniona Kontrolka Dane Tabelaryczne

Po kliknięciu ikony „+” (Dodaj) na karcie obiegu, **nowy wiersz kontrolki typu Dane Tabelaryczne** został automatycznie zainicjowany wartościami domyślnymi. Kolumny: **Nazwa** i **Ilość** oraz **Stawka VAT** zostały wypełnione wartościami zdefiniowanymi w zapytaniu SQL. Kolumny zostały wstępnie **wypełnione**, co skraca czas uzupełniania danych dla powtarzalnych pozycji. Dodatkowo dzięki rozbudowanemu zapytaniu SQL, dla nowo wprowadzonych wierszy istnieje możliwość **dwukierunkowego przeliczania wartości** (Netto, Brutto, Cena) co, ułatwia bieżącą aktualizację i kontrolę poszczególnych pozycji.

Pozycje dokumentu						
LP	Nazwa	Cena	Ilość	Netto	Stawka VAT	Brutto
1	a-vista		1,00		23%	
2	a-vista		1,00		23%	

Uzupełnione wiersze po dodaniu pozycji ikoną „+”

Rozpoczynasz pracę z Comarch BPM (dawniej DMS) i chcesz dowiedzieć się, jak korzystać z programu? A może masz już podstawową wiedzę o Comarch BPM (dawniej DMS) i chcesz

dowiedzieć się więcej?

[Sprawdź Szkolenia Comarch BPM!](#)

Czy wiesz, że w Comarch BPM można zweryfikować, czy dokument o danym numerze jest już wystawiony (tzw. weryfikacja numeru obcego)?

Zastosowanie

Dzięki tej konfiguracji system Comarch BPM zyskuje zdolność do **aktywnego wykrywania i ostrzegania** przed **duplikatami dokumentów** w obiegu. Zastosowanie to zapewnia **integralność i spójność danych**, uniemożliwiając wprowadzenie dokumentu, który już został zarejestrowany w systemie pod tym samym numerem i jest przypisany do tego samego kontrahenta. W przypadku wykrycia duplikatu, system natychmiast wyświetla **komunikat ostrzegawczy**.

Zawartość przykładu

- **Definicja Typu Obiegu (Workflow):** Wzorcowa ścieżka obiegu dokumentu kosztowego z krokami akceptacji i konfiguracją kontrolki **Komunikat**, konfiguracją kontrolki

Dane tabelaryczne i wygenerowaniem dokumentu do Comarch ERPXL

[Kliknij tutaj, aby pobrać przykład](#)

Zasada działania

- Mechanizm opiera się na kontrolce typu **Komunikat**, która pełni rolę **Obserwatora** dwóch kluczowych pól wejściowych: **Numer dokumentu** i **Kontrahent**.
- Kontrolka **Komunikat** jest wyzwalana przy każdej zmianie wartości w polach: **Numer dokumentu** oraz **Kontrahent**.
- Komunikat wykonuje **zapytanie SQL**, które przeszukuje bazę **BPM**, odwołując się do tabel dokumentów (**do.R0_Documents**) i wartości kontrolki (**do.R0_ControlsValueexpand**).
- Zapytanie szuka innego dokumentu (z pominięciem aktualnie edytowanego dokumentu: `t2.dwd_ID <> isnull(@^DocumentId @,0)`), w tym samym **Typie Obiegu** (`t1.DFH_Name='Weryfikacja numeru obcego BPM'`), o tym samym **Numerze dokumentu** (`t3.DWC_ValueString = @Numer_dokumentu@`) i przypisanego do tego samego **Kontrahenta** (`t4.DWC_ValueInt= @Kontrahent_Id@`).
- Jeśli duplikat zostanie znaleziony (`If exists (Select 1 from...)`), kontrolka wyświetla komunikat: „**Numer faktury dla wskazanego kontrahenta już istnieje**”.

Konfiguracja przykładu

1. Na karcie obiegu zdefiniowano następujące kontrolki niezbędne do skonfigurowania mechanizmu sprawdzania, czy dokument o danym numerze już istnieje w obiegu **Comarch BPM**:

- „**Numer dokumentu**” (identyfikator: **Numer_dokumentu**) – typ

Tekst

- „Kontrahent” (identyfikator: Kontrahent) – typ Kontrahent
- „Komunikat” (identyfikator: Komunikat) – typ Komunikat

2. Dodano obserwację kontrolek: Kontrolka **Komunikat** obserwuje pola „Numer dokumentu” i „Kontrahent”.


Komunikat


Nazwa (identyfikator)

Zmiana wartości (Obserwator)

Tytuł

Kontrolki powiązane

Numer dokumentu (Numer_dokumentu) 

Kontrahent (Kontrahent) 

[Dodaj](#)

Wzór na wartość

[Ustaw](#)

Kontrolka komunikat – kontrolki powiązane

3. Dodano zapytanie SQL: Kontrolka **Komunikat** zawiera zapytanie SQL, które wykonuje weryfikację, sprawdzając, czy inny dokument w tym samym **Typie Obiegu** (DFH_Name='Weryfikacja numeru obcego BPM') ma ten sam numer i tego samego kontrahenta.

- Jeśli duplikat zostanie znaleziony, wyświetla komunikat: „Numer faktury dla wskazanego kontrahenta już istnieje”.

```

BEGIN TRY
If exists (Select 1
from do.RO_ObjectsTypes as t1
join do.RO_Documents as t2 on t1.DFH_Id=t2.DFH_Id
join do.RO_ControlsValueexpand as t3 on t3.DWD_ID=t2.DWD_ID
join do.RO_ControlsValueExpand as t4 on t4.DWD_ID=t2.DWD_ID
where t3.DWC_ValueString = @Numer_dokumentu@ /*Odwołanie do
kontrolki Numer dokumentu*/
and t1.DFH_Name='Weryfikacja numeru obcego w BPM' /*Podstawić
właściwą nazwę typu obiegu*/
and t3.DKO_Name='Numer_dokumentu' /*Identyfikator kontrolki
Numer dokumentu*/
and t4.DKO_Name='Kontrahent' /*Identyfikator kontrolki
Kontrahent*/
and t4.DWC_ValueInt= @Kontrahent_Id@ /*Odwołanie do kontrolki
Kontrahent*/
and t2.dwd_ID <> isnull( @^DocumentId@,0) )
Begin

/*Informacja, która jest wyświetlana na karcie obiegu, w
przypadku duplikacji*/
SELECT 'Numer faktury dla wskazanego kontrahenta już
istnieje', 1
End
END TRY
BEGIN CATCH
DECLARE @ErrorMessage varchar(max) = 'W zapytaniu wystąpił
błąd : ' + ERROR_MESSAGE();
THROW 50001, @ErrorMessage ,1;
END CATCH

```

Komentarze dotyczące Logiki Zapytania:

- **t3.DWC_ValueString = @Numer_dokumentu@**: Odwołanie do kontrolki Numer dokumentu
- **t1.DFH_Name='Weryfikacja numeru obcego w BPM'**: Podstawić właściwą nazwę typu obiegu
- **t3.DKO_Name='Numer_dokumentu'**: Identyfikator kontrolki Numer dokumentu

- **t4.DKO_Name='Kontrahent'**: Identyfikator kontrolki Kontrahent
- **t4.DWC_ValueInt= @Kontrahent_Id@**: Odwołanie do kontrolki Kontrahent
- **SELECT 'Numer faktury dla wskazanego kontrahenta już istnieje', 1**: Informacja, która jest wyświetlana na karcie obiegu, w przypadku duplikacji.
- **Wartość 1 (Blokada Zapisu)**: Jeżeli zapytanie zwraca wartość **1**, komunikat pojawia się na karcie obiegu, a zapisanie nowoutworzonego dokumentu lub przekazanie już zapisanego dokumentu do kolejnego etapu zostanie **zablokowane**. System wymusi poprawienie danych przez użytkownika, uniemożliwiając wprowadzenie duplikatu do obiegu.
- **Wartość 0 (Brak Blokad)**: Jeżeli zamiast jedynki zwracana jest wartość **0**, komunikat pojawi się jedynie jako **ostrzeżenie**. Użytkownik będzie mógł **zapisać i procesować** dokument dalej, ponieważ system zasygnalizuje potencjalny duplikat, ale nie zablokuje pracy.

Uwaga

W kontekście weryfikacji duplikatu, użycie **wartości 1** jest standardowe i konieczne, aby mechanizm ten skutecznie **uniemożliwił** wprowadzenie powtarzających się dokumentów

Wyrażenie kontrolki

Uwaga: Zmiana nazwy kontrolki może spowodować nieprawidłowe działanie wyrażenia.

```

1 BEGIN TRY
2 IF EXISTS (SELECT 1
3 FROM do_RO_ObjectsTypes AS t1
4 JOIN do_RO_Documents AS t2 ON t1.DFH_Id=t2.DFH_Id
5 JOIN do_RO_ControlsValueExpand AS t3 ON t3.DwD_ID=t2.DwD_ID
6 JOIN do_RO_ControlsValueExpand AS t4 ON t4.DwD_ID=t2.DwD_ID
7 WHERE t3.DwC_ValueString = @Numer_dokumentu@ /*Odwołanie do kontrolki Numer dokumentu*/
8 AND t1.DFH_Name='Weryfikacja numeru obcego w BPM' /*Podstawic właściwą nazwę typu obiegu*/
9 AND t3.DKO_Name='Numer dokumentu' /*Identyfikator kontrolki Numer dokumentu*/
10 AND t4.DwC_ValueInt= @Kontrahent@ /*Identyfikator kontrolki Kontrahent*/
11 AND t4.DwC_ValueInt= @Kontrahent_Id@ /*Odwołanie do kontrolki Kontrahent*/
12 AND t2.dwd_ID <> ISNULL( @^DocumentId@,0) )
13 BEGIN
14 /*Informacja, która jest wyświetlana na karcie obiegu, w przypadku duplikacji*/
15 SELECT 'Numer faktury dla wskazanego kontrahenta już istnieje', 1
16 END
17
18 END TRY
19 BEGIN CATCH
20 DECLARE @ErrorMessage varchar(max) = 'W zapytaniu wystąpił błąd : ' + ERROR_MESSAGE();
21 THROW 50001, @ErrorMessage, 1;
22 END CATCH
23

```

Kontrolki

- ^DocumentCompanyId
- ^UserId
- ^DocumentId

Składnia:

```

SELECT varchar(max), [int]
Varchar - treść komunikatu
Int - blokada przekazania do kolejnego etapu (0 - brak blokady, 1 -
ustawiono blokadę). Parametr opcjonalny, domyślnie ustawiona
wartość 0.

```

Zapisz Zamknij

[Komunikat] Weryfikacja numeru obcego

Komunikat

Nazwa (identyfikator) Komunikat

Zmiana wartości (Obserwator)

Tytuł Weryfikacja Numeru Obcego

Kontrolki powiązane Numer dokumentu (Numer dokumentu)

Kontrahent (Kontrahent)

Dodaj

Wzór na wartość SQL OD

Ustaw

Zapytanie SQL w kontrolce Komunikat

Weryfikacja Duplikacji Dokumentów

Na nowoutworzonym dokumencie, który nie został jeszcze zapisany, po wprowadzeniu numeru dokumentu do kontrolki „Numer dokumentu” i następnie wybraniu kontrahenta w kontrolce „Kontrahent”, mechanizm weryfikacji zostaje natychmiast wyzwolony. System wykonał zapytanie SQL, sprawdzając bazę BPM. Kluczowym elementem zapytania jest warunek **t2.dwd_ID <> ISNULL(@^DocumentId@,0)**, który zapewnia, że weryfikacja ignoruje bieżący dokument (ponieważ dopiero jest tworzony i nie ma nadanego numeru w obiegu BPM, który by go identyfikował jako zapisany). W tym momencie system wykrył, że dokument o tym **samej numerze** dla wskazanego **kontrahenta** już istnieje w BPM i pojawia się komunikat ostrzegawczy: „Numer faktury dla wskazanego kontrahenta już istnieje”. Wartość **1** w instrukcji SELECT zapytania oznacza **Blokadę zapisu**. System **uniemożliwia** zapisanie dokumentu do kolejnego etapu, dopóki nie zostanie skorygowany numer dokumentu lub kontrahent.



Data utworzenia : 09-10-20

Dokumenty

Dane ogólne

Skan dokumentu



Numer dokumentu

FS12/2025

Data wystawienia

9.10.2025

Data zakupu

9.10.2025

Data wpływu

9.10.2025

Wybór kontrahenta

ALOZA | Aloza S.A | 5678765432

Kontrahent



[ALOZA] Aloza S.A

WeryfikacjaNumeruObcego



Numer faktury dla wskazanego kontrahenta już istnieje

OK

Komunikat informujący o duplikacji

Rozpoczynasz pracę z Comarch BPM (dawniej DMS) i chcesz dowiedzieć się, jak korzystać z programu? A może masz już podstawową wiedzę o Comarch BPM (dawniej DMS) i chcesz dowiedzieć się więcej?

[Sprawdź Szkolenia Comarch BPM!](#)

[Powrót do początku artykułu](#)

Czy wiesz, że w kontrolce Dane tabelaryczne wartości w poszczególnych kolumnach mogą się przeliczać w oparciu o zmianę wartości w innej kolumnie przy pomocy zapytania SQL?

Zastosowanie

Dzięki zastosowaniu tej konfiguracji, proces wprowadzania danych w Comarch BPM zyskuje pełną zdolność **automatycznego** kalkulowania wszędzie tam, gdzie użyta jest kontrolka tabelaryczna. Głównym celem jest **eliminacja ręcznych obliczeń**, co bezpośrednio przekłada się na mniejszą liczbę błędów i znacznie **szybszą pracę użytkownika**. Konfiguracja ta umożliwia automatyczne przeliczanie wartości w kolumnach wynikowych (np. **Netto**, **Brutto**) w oparciu o zmianę danych wejściowych (np. **Cena** lub **Ilość**) w tym samym wierszu.

Film obrazujący zastosowanie konfiguracji

<https://pomoc.comarch.pl/bpm/wp-content/uploads/2025/10/filmauto.mp4>

Zawartość przykładu

- **Definicja Typu Obiegu (Workflow):** Wzorcowa ścieżka obiegu dokumentu kosztowego z krokami akceptacji i konfiguracją kontrolki Dane tabelaryczne i wygenerowaniem dokumentu do Comarch ERPXL

[Kliknij tutaj, aby pobrać przykład](#)

Zasada działania

Mechanizm opiera się na ciągłym **monitorowaniu zmian wartości w kontrolce tabelarycznej** i automatycznym uruchamianiu zapytania SQL, które przelicza i aktualizuje wartości w kolumnach wynikowych.

- Kontrolka Dane Tabelaryczne jest skonfigurowana tak, aby **obserwować samą siebie** i reagować na każdą edycję wiersza.
- **Zapytanie SQL**, zaimplementowane na zakładce **Inicjowanie wartości**, jest wyzwalane przy każdej zmianie wartości w kolumnach wejściowych (**Cena**, **Ilość** lub **Stawka VAT**).
- **Przeliczania W Przód: (Cena/Ilość):** Logika **IF** w kodzie SQL jest wyzwalana przy zmianie kolumn wejściowych (np. **Cena** lub **Ilość**). W tym trybie **Netto** i **Brutto** są obliczane na podstawie wartości wejściowych i **stawki VAT**.
- **Przeliczania W Tył: (Brutto):** Logika **ELSE IF** jest wyzwalana przy zmianie kolumny **Brutto**. W tym trybie system przelicza i aktualizuje wartość w kolumnie **Cena** lub **Netto**, traktując Ilość jako stałą, zachowując tym samym wewnętrzną spójność wiersza.
- **Zapytanie SQL** pobiera niezbędne dane (np. wartość **Stawki VAT** z tabeli **CDN.Nazwy**). Następnie, za pomocą komendy

UPDATE, bezpośrednio przelicza i aktualizuje wartości w kolumnach wynikowych (**Netto** oraz **Brutto**).

- Kolumna wynikowa **Netto** została ustawiona w trybie „Do odczytu”, aby użytkownik nie mógł ręcznie jej edytować.
- Osobna kontrolka typu **Podsumowanie** obserwuje kolumnę wynikową (**Brutto**) i zlicza jej sumę za pomocą prostego zapytania SQL (`select sum (Brutto) from @Elementy@`).

Konfiguracja przykładu

1. **Utworzono kontrolkę Tabelaryczną:** Dodano kontrolkę typu **Dane tabelaryczne** o nazwie (identyfikatorze) **Elementy** i skonfigurowano w niej następujące kolumny:

- **„Nazwa”** (identyfikator kolumny: **Nazwa**) – typ **Tekst**
- **„Cena”** (identyfikator kolumny: **Cena**) – typ **Liczba stałoprzecinkowa**
- **„Ilość”** (identyfikator kolumny: **Ilosc**) – typ **Liczba stałoprzecinkowa**
- **„Netto”** (identyfikator kolumny: **Netto**) – typ **Liczba stałoprzecinkowa**
- **„Stawka VAT”** (identyfikator kolumny: **StawkaVAT**) – typ **Lista**
- **„Brutto”** (identyfikator kolumny: **Brutto**) – typ **Liczba stałoprzecinkowa**

Pozycje dokumentu

Elementy

LP | Nazwa

| Cena

| Ilość

| Netto

| Stawka VAT

| Brutto



Kolumny w kontrolce typu Dane tabelaryczne

2. W kontrolce typu **Dane tabelaryczne** na zakładce **Listy** zdefiniowano zapytanie pobierające:

▪ **Stawki VAT:**

```
select Naz_gidLp as Id,  
CASE  
  WHEN Naz_Nazwa='A 23.00' THEN '23%'  
  WHEN Naz_Nazwa='B 8.00' THEN '8%'  
  WHEN Naz_Nazwa='C 0.00' THEN '0%'  
  WHEN Naz_Nazwa='D 0.00' THEN 'ZW'  
  WHEN Naz_Nazwa='E 0.00' THEN 'NP'  
  WHEN Naz_Nazwa='F 7.00' THEN '7%'  
  WHEN Naz_Nazwa='G 5.00' THEN '5%'  
END  
from CDN.Nazwy  
where Naz_GIDTyp = 624 AND trim(substring(Naz_Nazwa,9,2))= ''
```

Dane tabelaryczne

OGÓLNE

LISTY

INICJOWANIE WARTOŚCI

OGRANICZENIE EDYCJI

Lista 1



Nazwa

Stawka VAT

Inicjowanie

SQL ERP



Rodzaj identyfikatora

Int



[Pokaż](#) [Usuń](#)

Tryb pracy

Lista



Lista Stawka VAT w kontrolce Dane tabelaryczne

3. **Dodano obserwację kontrolki:** W kontrolce typu **Dane tabelaryczne**, na zakładce **Inicjowanie wartości**, dodano powiązanie do samej siebie.

Dane tabelaryczne

OGÓLNE

LISTY

INICJOWANIE WARTOŚCI

OGRANICZENIE EDYCJI

Inicjowanie wartości

Inicjowanie

Brak

Kontrolki powiązane

Elementy (Elementy)



Dodaj

Wzór na wartość

SQL ERP

Ustaw

Powiązanie kontrolki do samej siebie

4. **Dodano zapytanie SQL:** W kontrolce typu **Dane tabelaryczne**, na zakładce **Inicjowanie wartości** w sekcji **Wzór na wartość SQL ERP** dodano zapytanie, które pobiera stawki VAT z bazy XL i na podstawie **Ilości** i **Ceny** wylicza **wartość Netto**, po uzupełnieniu **Stawki VAT** wylicza **wartość Brutto**, a po zmianie wartości **Brutto** wylicza **Cenę** i **wartość Brutto**.

```
/*Deklaracja Stawki VAT */
DECLARE @VAT AS DECIMAL(4, 2)
```

```
/* Pobranie stawki VAT z tabeli CDN.Nazwy na podstawie kolumny
StawkaVAT w bieżącym wierszu */
SELECT @VAT = CAST(SUBSTRING(Naz_Nazwa1, 0, CHARINDEX(' ',
Naz_Nazwa1)) AS DECIMAL(4, 2)) / 100
```

```
FROM CDN.Nazwy
WHERE Naz_GIDTyp = 624
AND Naz_GIDLp = (
SELECT [StawkaVAT]
FROM @Elementy@
WHERE POS = @Elementy_Row@
)
```

```
/* JEŚLI ZMIENIONO KOLUMNY: 2 (Cena), 3 (Ilość), 5 (Stawka
VAT) */
```

```
IF @Elementy_Column@=2 OR @Elementy_Column@=3 OR
@Elementy_Column@=5
```

```
BEGIN
```

```
/* PRZELICZANIE W PRZÓD (Cena/Ilość/Stawka VAT ->
Netto/Brutto) */
```

```
/*Wyliczenie i aktualizacja netto i brutto */
```

```
UPDATE @Elementy@
```

```
SET
```

```
[Netto] = (Ilosc * Cena),
```

```
[Brutto] = (Ilosc * Cena) * (@VAT + 1)
```

```
WHERE POS = @Elementy_Row@
```

```
/* !!! OBOWIĄZKOWE: Odświeżenie wiersza !!! */
```

```
SELECT * FROM @Elementy@
```

```
END
```

```
/* JEŚLI ZMIENIONO KOLUMNĘ: 6 (Brutto) */
```

```
IF @Elementy_Column@=6
```

```
BEGIN
```

```
/* PRZELICZANIE W TYŁ (Brutto -> Cena/Netto) */
```

```
/* Aktualizacja ceny i netto (Ilość stała) */
```

```
UPDATE @Elementy@
```

```
SET
```

```
[Cena] = [Brutto] / (@VAT + 1), -- Obliczenie Ceny Netto z
Brutto
```

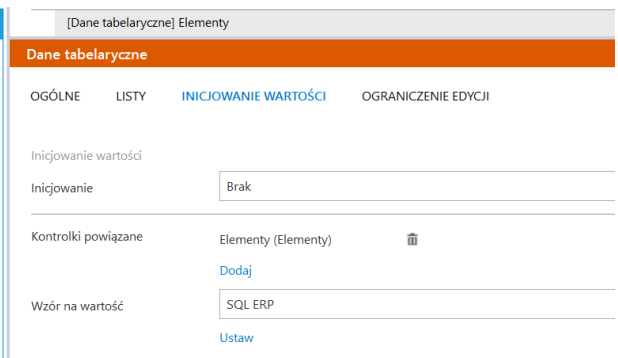
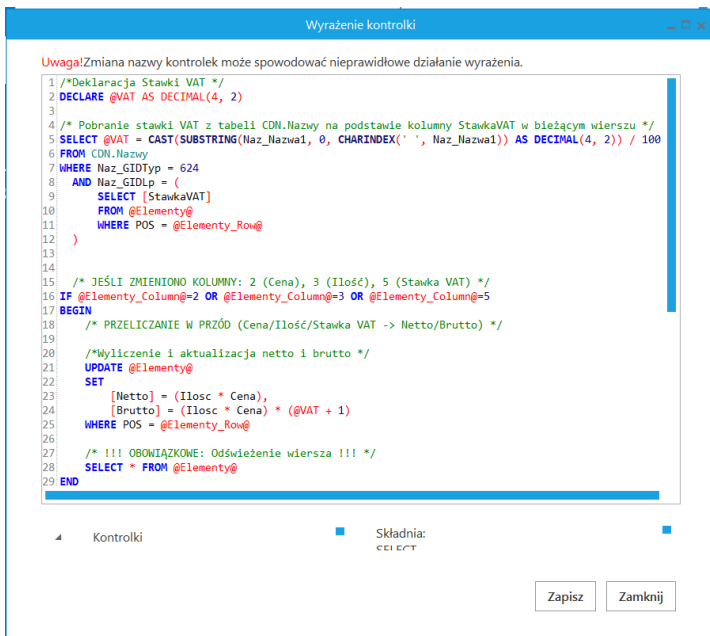
```
[Netto] = [Ilosc] * ([Brutto] / (@VAT + 1)) -- Obliczenie
Wartości Netto
```

```
WHERE POS = @Elementy_Row@
```

```

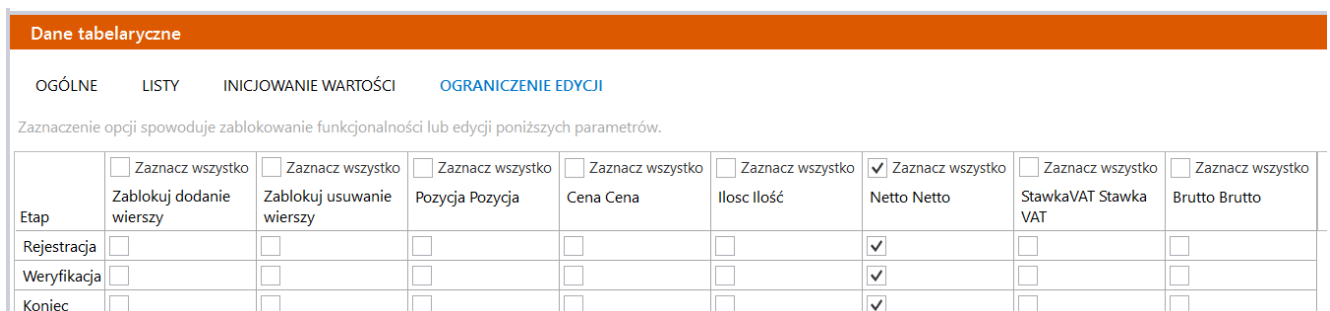
/* !!! OBOWIĄZKOWE: Odświeżenie wiersza !!! */
SELECT * FROM @Elementy@
END

```



Zapytanie SQL w kontrolce Dane tabelaryczne na zakładce Inicjowanie wartości

5. Tryb pracy: W kontrolce typu Dane tabelaryczne, na zakładce **Ograniczenie edycji** kolumnę „Netto” ustawiono w trybie „Do odczytu”.



Kolumna Netto w trybie „Do odczytu”

6. **Utworzono kontrolkę liczbową:** Dodano kontrolkę typu **Liczba stałoprzecinkowa** o nazwie (identyfikatorze) **Podsumowanie**, w której dodano powiązanie do kontrolki **Elementy** i skonfigurowano w niej zapytanie, które zlicza wartość z kolumny **Brutto** z kontrolki tabelarycznej:

```
select sum (Brutto) from @Elementy@
```

The screenshot shows the configuration interface for a 'Liczba stałoprzecinkowa' (Decimal Number) control. The interface is split into two main panels.

Left Panel: Wyrażenie kontrolki (Control Expression)

- Warning: 'Uwaga! Zmiana nazwy kontrolki może spowodować nieprawidłowe działanie wyrażenia.' (Warning! Changing the control name may cause incorrect expression operation.)
- SQL Expression: `1 select sum (Brutto) from @Elementy@`
- Control List: Shows 'Kontrolki' with '^DocumentCor' and '^UserId'.
- Syntax: 'Składnia: SELECT decimal (22,4) decimal – wartość stałoprzecinkowa. Przykład: BEGIN TRY'
- Buttons: 'Zapisz' (Save) and 'Zamknij' (Close)

Right Panel: Liczba stałoprzecinkowa (Decimal Number Control Configuration)

- Info: 'Kontrolkę można uzupełnić liczbą złożoną z maksymalnie 18 cyfr przed przecinkiem i 6 miejsc po przecinku' (The control can be filled with a number consisting of a maximum of 18 digits before the decimal point and 6 digits after the decimal point.)
- Fields: 'Nazwa wyświetlana' (Podsumowanie), 'Nazwa (identyfikator)' (Podsumowanie), 'Liczba miejsc po przecinku, do której będzie zaokrąglana wartość' (2), 'Prezentuj na całej szerokości' (checkbox), 'Ustaw kontrolkę na początku wiersza' (checkbox), 'Pokaż na liście' (checkbox).
- Value Initialization: 'Inicjowanie wartości' (Brak), 'Inicjowanie' (Brak), 'Dokument', 'Pole'.
- Observer: 'Zmiana wartości (Obserwator)'
- Linked Controls: 'Kontrolki powiązane' (Elementy (Elementy) with a trash icon), 'Dodaj' (Add).
- Value Pattern: 'Wzór na wartość' (SQL OD), 'Pokaż' (Show), 'Usuń' (Delete).

Kontrolka Liczbowa Podsumowanie

Uzupełniona Kontrolka Dane Tabelaryczne

Po wprowadzeniu wartości w kontrolce typu **Dane tabelaryczne** w kolumnach **Cena**, **Ilość** lub **Stawka VAT** kontrolka typu **Dane Tabelaryczne** została **automatycznie zaktualizowana**. Dla każdej pozycji, system automatycznie przeliczył i uzupełnił wartości w kolumnach **Netto** oraz **Brutto** w tym samym wierszu. Możliwe jest przeliczanie wartości **w przód** (Cena/Ilość → Netto/Brutto) oraz **w tył** (Brutto → Cena/Netto). Wartości w kolumnach

wynikowych (**Netto**, **Brutto**) są widoczne, ale kolumna **Netto** jest zablokowana do ręcznej edycji (tryb „Do odczytu”), co gwarantuje poprawność danych. Kontrolka **Podsumowanie** wyświetla poprawną sumę końcową z kolumny **Brutto** kontrolki typu Dane tabelaryczne.

Karta obiegu						
Elementy						
LP	Pozycja	Cena	Ilość	Netto	Stawka VAT	Brutto
1	T1	3,00	4,00	12,00	23%	14,76
2	T2	6,00	8,00	48,00	23%	59,04
3	T3	90,00	4,00	360,00	8%	388,80
+						
✕						
Podsumowanie						
		462,60				

Kontrolka typu Dane tabelaryczne z uzupełnionymi kolumnami

Rozpoczynasz pracę z Comarch BPM (dawniej DMS) i chcesz dowiedzieć się, jak korzystać z programu? A może masz już podstawową wiedzę o Comarch BPM (dawniej DMS) i chcesz dowiedzieć się więcej?

[Sprawdź Szkolenia Comarch BPM!](#)

Czy wiesz, że kontrolkę Dane tabelaryczne można zainicjować wartością domyślną?

Zastosowanie

Dzięki tej konfiguracji, proces obsługi dokumentów w Comarch BPM zyskuje bezpośrednią oszczędność czasu i gwarancję spójności danych. Umożliwia automatyczne wstępne wypełnienie kontrolki tabelarycznej na wskazanym etapie obiegu, eliminując potrzebę ręcznego dodawania wierszy. W efekcie, użytkownik zyskuje natychmiastową gotowość do pracy w momencie rozpoczęcia obiegu. Minimalizuje to błędy poprawności danych, gwarantując, że dane są dokładne, spójne i aktualne oraz znacząco przyspiesza obsługę standardowych, powtarzalnych procesów.

Zawartość przykładu

- **Definicja Typu Obiegu (Workflow):** Wzorcowa ścieżka obiegu, która zawiera konfigurację kontrolki Dane tabelaryczne

[Kliknij tutaj, aby pobrać przykład](#)

Zasada działania

Mechanizm wykorzystuje zasadę, że **kontrolka tabelaryczna to tabela tymczasowa SQL**, którą można wypełnić dowolnymi danymi za pomocą zapytania.

- **Zapytanie SQL** jest umieszczone na zakładce **Inicjowanie wartości** i zawiera dane, które mają pojawić się w tabeli.
- Na etapie, na którym kontrolka ma zostać zainicjowana, zaznacza się opcję „**Inicjalizacja SQL**”. Zapytanie jest uruchamiane raz, wypełniając wiersze wartością domyślną.
- Aby zainicjować kontrolkę więcej niż jednym wierszem, wykorzystuje się instrukcję operacji na zbiorach **UNION**, która łączy wyniki poszczególnych zapytań **SELECT**.
- Ostatnią kolumną w zapytaniu jest kolumna **LP (numer wiersza)**, którą w zapytaniu inicjalizujemy zerami (0 as LP). System automatycznie zastępuje te zera kolejnymi wartościami porządkowymi.

Konfiguracja przykładu

1. **Utworzono kontrolkę Tabelaryczną:** Dodano kontrolkę typu **Dane tabelaryczne** o nazwie (identyfikatorze) **Dokumenty** i skonfigurowano w niej następujące kolumny:

- **"Nazwa"** (identyfikator kolumny: **Nazwa**) – typ **Tekst**
- **"Kwota"** (identyfikator kolumny: **Kwota**) – typ **Liczba stałoprzecinkowa**
- **"Termin"** (identyfikator kolumny: **Termin**) – typ **Data i godzina**

Karta obiegu

Dokumenty

LP	Nazwa	Kwota	Termin
----	-------	-------	--------



Kolumny w kontrolce Dane tabelaryczne

2. **Dodano zapytanie inicjujące SQL:** W kontrolce **Dane tabelaryczne**, na zakładce **Inicjowanie wartości** w sekcji **Inicjowanie SQL** dodano zapytanie, które uzupełni kontrolkę wartościami domyślnymi: **numerem faktury, kwotą oraz terminem.**

```
SELECT
```

```
        'Faktura  
001' AS nazwa, 1500.00 AS kwota, '2025-04-15' AS termin,  
0 as LP
```

```
UNION
```

```
SELECT
```

```
        'Faktura 002' AS nazwa, 3200.50 AS kwota, '2025-04-20'  
AS termin, 0 as LP
```

Zapytanie SQL

Uwaga! Wprowadzenie niepoprawnych danych może powodować błędy aplikacji.
Treść zapytania:

```

1 SELECT
2   'Faktura 001' AS nazwa, 1500.00 AS kwota, '2025-04-15' AS termin, 0 as LP
3 UNION
4 SELECT
5   'Faktura 002' AS nazwa, 3200.50 AS kwota, '2025-04-20' AS termin, 0 as LP

```

Kontrolki

- ^UserId
- ^DocumentId
- ^ParentDocumentId
- ^WorkflowId
- ^StageId

Składnia:
SELECT
varchar(900) - Nazwa
decimal(22, 4) - Kwota
'YYYYMMDD' - Termin
int - tryb: 1 - nadpisanie
wierszy, 2 - dodanie wierszyPOS -
pozycja (opcjonalny numer wiersza)

[Dane tabelaryczne] Dokumenty

Dane tabelaryczne

OGÓLNE LISTY **INICJOWANIE WARTOŚCI** OGRANICZENIE EDYCJI

Inicjowanie wartości

Inicjowanie SQL OD ▼

Pokaż Usuń

Kontrolki powiązane Dodaj

Wzór na wartość SQL OD ▼

Ustaw

Zapytanie SQL w kontrolce Dane tabelaryczne na zakładce Inicjowanie wartości

3. Włączenie Inicjalizacji: Na etapie, na którym kontrolka ma się zainicjować wartościami domyślnymi (np. na pierwszym etapie obiegu), zaznaczono opcję „Inicjalizacja SQL”.

Wprowadzenie

Weryfikacja

Archiwizacja

Etap

OGÓLNE UPRAWNIENIA PRZEKAZANIE MAIL **KONTROLKI** PARAMETRY


Włącz funkcję obserwatora podczas otwierania dokumentu

Kontrolka	<input checked="" type="checkbox"/> Zaznacz wszystko Widoczna	<input type="checkbox"/> Zaznacz wszystko Tylko do odczytu	<input type="checkbox"/> Zaznacz wszystko Wymagana	<input checked="" type="checkbox"/> Zaznacz wszystko Inicjalizacja Sql/C#	<input type="checkbox"/> Zaznacz wszystko Odświeżanie tabeli na żądanie	Kolejność inicjowania ▼
Zakładka 1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	▲
Karta obiegu	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	▲
Dokumenty	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	▼







Włączenie Inicjalizacji SQL



Uzupełniona Kontrolka typu Dane Tabełaryczne

Po utworzeniu dokumentu, kontrolka typu **Dane Tabełaryczne** została **automatycznie zainicjowana domyślnymi** wartościami pobranymi z zapytania SQL. **Tabela zawiera wiersze z kompletnymi danymi: numer faktury, kwotą, terminem**, eliminując tym samym konieczność **ręcznego** dodawania pozycji i **skracając czas** obsługi dokumentu.

Karta obiegu 

Dokumenty

LP	Nazwa	Kwota	Termin
1	Faktura 001	1 500,00  	15.04.2025 
2	Faktura 002	3 200,50  	20.04.2025 

Kontrolka typu Dane tabelaryczne z uzupełnionymi kolumnami

Rozpoczynasz pracę z Comarch BPM (dawniej DMS) i chcesz dowiedzieć się, jak korzystać z programu? A może masz już podstawową wiedzę o Comarch BPM (dawniej DMS) i chcesz dowiedzieć się więcej?

[Sprawdź Szkolenia Comarch BPM!](#)

Czy wiesz, że można wypełnić kontrolkę typu Dane tabelaryczne pozycjami dokumentu zacytowanego przez Comarch OCR oraz przeliczyć wartości przy współpracy z Comarch ERP XL?

Zastosowanie

Dzięki zastosowaniu tej zintegrowanej konfiguracji, proces obiegu faktur kosztowych w Comarch BPM zyskuje pełną automatyzację i spójność danych. Głównym celem jest:

- **Eliminacja ręcznego wprowadzania pozycji dokumentów:** Dane zacytowane przez Comarch OCR są automatycznie mapowane i wypełniają kontrolkę tabelaryczną na karcie obiegu.
- **Automatyczne kalkulowanie wartości:** Kontrolka tabelaryczna uzyskuje pełną zdolność automatycznego przeliczania wszędzie tam, gdzie zmienione zostaną dane wejściowe (np. Cena, Ilość, Stawka VAT, Brutto).
- **Minimalizacja błędów i skrócenie czasu obsługi faktury:** Proces akceptacji bazuje na poprawnych i spójnych danych, które są automatycznie kalkulowane i aktualizowane, zanim dokument zostanie zaksięgowany w ERP XL.

Zawartość przykładu

- **Definicja Typu Obiegu (Workflow):** Wzorcowa ścieżka obiegu dokumentu kosztowego z krokami akceptacji, konfiguracją kontrolki Dane tabelaryczne i wygenerowaniem dokumentu do Comarch ERPXL
- **Punkt konfiguracyjny ACD:** Wzorcowa konfiguracja, która jest odpowiedzialna za pobranie ID dokumentu OCR, Załącznika oraz wszystkich kluczowych danych nagłówkowych (NIP, Kontrahent, Dane kontrahenta, Data zakupu, Data wystawienia, Termin płatności, Forma płatności, Waluta) oraz inicjację dokumentu obiegowego.

[Kliknij tutaj, aby pobrać przykład](#)

Zasada działania

Mechanizm opiera się na **dwóch trybach działania** zintegrowanych w jednej kontrolce **Dane Tabelaryczne („Elementy”)**:

1. Pobranie pozycji z Comarch OCR (Wyzwalacz: ID_OCR):

- Po przetworzeniu dokumentu przez Comarch OCR, punkt ACD zapisuje **ID dokumentu OCR** do dedykowanej kontrolki typu Liczba całkowita o nazwie **ID_OCR**.
- Kontrolka tabelaryczna „Elementy” obserwuje to pole, reagując na jego wypełnienie.
- **Zapytanie SQL** (umieszczone na zakładce Inicjowanie Wartości) jest filtrowane za pomocą parametru **@ID_OCR@** i pobiera wiersze z tabeli **do.OC_Document_Items** w bazie BPM.
- Na podstawie zwróconych wierszy kontrolka **DT** jest

automatycznie wypełniana pozycjami faktury.

- W zapytaniu OCR używana jest logika CASE WHEN do konwersji wartości tekstowych (**Stawki VAT, Jednostki miary**) na **identyfikatory liczbowe**, wymagane przez BPM.

2. Automatyczne Przeliczanie (Wyzwalacz: Edycja wiersza DT):

- Kontrolka **DT** jest skonfigurowana tak, aby **obserwować samą siebie**.
 - Zapytanie SQL jest wyzwalane przy każdej zmianie wartości w kolumnach wejściowych (**Cena, Ilość lub Stawka VAT**).
 - **Przeliczanie W Przód**: Logika IF w kodzie SQL jest wyzwalana przy zmianie Ceny, Ilości lub Stawki VAT, obliczając **Netto** i **Brutto**.
 - **Przeliczanie W Tył**: Logika ELSE IF jest wyzwalana przy zmianie kolumny **Brutto**, przeliczając i aktualizując wartość **Ceny** lub **Netto** (przy zachowaniu stałej Ilości).
 - Zapytanie to wykorzystuje komendę **UPDATE** do bezpośredniego przeliczania i aktualizacji wartości w kolumnach wynikowych (**Netto, Brutto**)
- Osobna kontrolka **WartoscNetto** obserwuje kolumnę wynikową (**Netto**) i zlicza jej sumę za pomocą prostego zapytania SQL (select sum (Netto) from @Elementy@).
- Osobna kontrolka **WartoscBrutto** obserwuje kolumnę wynikową (**Brutto**) i zlicza jej sumę za pomocą prostego zapytania SQL (select sum (Brutto) from @Elementy@).

Konfiguracja przykładu

1. Utworzono kontrolkę **ID_OCR**: Na karcie obiegu dodano kontrolkę typu **Liczba całkowita** o nazwie (identyfikatorze) **ID_OCR**, do której będzie zapisywane ID OCR-owanego dokumentu.

Techniczne



[Liczba całkowita] ID_OCR

Liczba całkowita

 Kontrolkę można uzupełnić liczbą z zakresu od -2 147 483 648 do 2 147 483 647

Nazwa wyświetlana

ID_OCR

Nazwa (identyfikator)

ID_OCR

Prezentuj na całej szerokości

Ustaw kontrolkę na początku wiersza

Pokaż na liście

Kontrolka Liczba całkowita ID_OCR

2. Utworzono kontrolkę Tabelaryczną: Dodano kontrolkę typu **Dane tabelaryczne** o nazwie (identyfikatorze) **Elementy** i skonfigurowano w niej następujące kolumny:

- **"Nazwa" (identyfikator kolumny: Nazwa)** – typ **Tekst**
- **"Cena" (identyfikator kolumny: Cena)** – typ **Liczba stałoprzecinkowa**
- **"Ilość" (identyfikator kolumny: Ilosc)** – typ **Liczba stałoprzecinkowa**
- **"Jednostka miary" (identyfikator kolumny: JM)** – typ **Lista**
- **"Netto" (identyfikator kolumny: Netto)** – typ **Liczba stałoprzecinkowa**
- **"Stawka VAT" (identyfikator kolumny: StawkaVAT)** – typ **Lista**
- **"Brutto" (identyfikator kolumny: Brutto)** – typ **Liczba**

stałoprzecinkowa

Pozycje dokumentu

Elementy

LP	Nazwa	Cena	Ilość	Jednostka miary	Netto	Stawka VAT	Brutto
----	-------	------	-------	-----------------	-------	------------	--------



Kolumny w kontrolce Dane tabelaryczne

3.W kontrolce typu **Dane tabelaryczne** na zakładce **Listy** zdefiniowano zapytanie pobierające:

▪ Stawki VAT:

```
select Naz_gidLp as Id,  
CASE  
  WHEN Naz_Nazwa='A 23.00' THEN '23%'  
  WHEN Naz_Nazwa='B 8.00' THEN '8%'  
  WHEN Naz_Nazwa='C 0.00' THEN '0%'  
  WHEN Naz_Nazwa='D 0.00' THEN 'ZW'  
  WHEN Naz_Nazwa='E 0.00' THEN 'NP'  
  WHEN Naz_Nazwa='F 7.00' THEN '7%'  
  WHEN Naz_Nazwa='G 5.00' THEN '5%'  
END  
from CDN.Nazwy  
where Naz_GIDTyp = 624 AND trim(substring(Naz_Nazwa,9,2))= ''
```

Dane tabelaryczne

OGÓLNE

LISTY

INICJOWANIE WARTOŚCI

OGRANICZENIE EDYCJI

Lista 1



Nazwa

Stawka VAT

Inicjowanie

SQL ERP

Rodzaj identyfikatora

Int

[Pokaż](#) [Usuń](#)

Tryb pracy

Lista

Lista Stawka VAT w kontrolce Dane tabelaryczne


▪ Jednostki miary:

```
SELECT Naz_GIDLp,Naz_Nazwa from cdn.Nazwy where Naz_GIDTyp=144  
;
```


[Dane tabelaryczne] Elementy


Dane tabelaryczne

OGÓLNE LISTY INICJOWANIE WARTOŚCI OGRANICZENIE EDYCJI


Lista 1 

Nazwa JM

Inicjowanie Jednostki miary 

Rodzaj identyfikatora Int 

[Pokaż](#) [Usuń](#)

Tryb pracy Lista 

[Dodaj](#)

Lista Jednostka miary w kontrolce Dane tabelaryczne

4. Dodano obserwację kontrolki: W kontrolce **Dane tabelaryczne**, na zakładce **Inicjowanie wartości**, dodano powiązanie do kontrolki **ID_OCR** oraz **do samej siebie**.

[Dane tabelaryczne] Elementy



Dane tabelaryczne

OGÓLNE LISTY **INICJOWANIE WARTOŚCI** OGRANICZENIE EDYCJI

Inicjowanie wartości

Inicjowanie

Kontrolki powiązane

ID_OCR (ID_OCR)	
Elementy (Elementy)	

[Dodaj](#)

Wzór na wartość

[Ustaw](#)

Powiązanie do kontrolki ID_OCR i samej siebie

5. Zintegrowane zapytanie SQL do obsługi obu trybów: W kontrolce typu **Dane tabelaryczne**, na zakładce **Inicjowanie wartości** w sekcji **Wzór na wartość SQL ERP** dodano zapytanie, które pobiera dane z tabel **do.OC_Document_Items** oraz **do.OC_Documents** na podstawie przekazanego parametru **@ID_OCR@** (ID dokumentu OCR). W zapytaniu SQL konieczne jest użycie logiki CASE WHEN do konwersji wartości tekstowych **jednostek miary** ('szt') i **stawek VAT** (np. '23.00') na odpowiednie identyfikatory liczbowe akceptowane przez Comarch BPM. Dodano również zapytanie, które na podstawie **Ilości** i **Ceny** wylicza **wartość Netto**, po zmianie **Stawki VAT** wylicza **wartość Brutto**, a po zmianie wartości **Brutto** wylicza **Cenę** i **wartość Netto**.

Uwaga

Należy upewnić się, że identyfikatory liczbowe (ID) **stawek VAT** i **jednostek miary** użyte w zapytaniu SQL są zgodne z tymi, które aktualnie obowiązują w bazie Comarch ERP XL współpracującej z Comarch BPM.

Uwaga

W zapytaniu przed nazwą tabel **do.OC_Document_Items** oraz **do.OC_Documents** należy wskazać właściwą nazwę bazy **Comarch BPM**.

Ta sekcja jest wywoływana tylko raz, podczas inicjowania obiegu z OCR:

```
If @^SenderControlName@ = 'ID_OCR'
```

```
Begin
```

```
Select
```

```
ODI_Name, /*Nazwa*/
```

```
ODI_NetUnitPrice, /*Cena netto*/
```

```
ODI_Count, /*Ilość*/
```

```
CASE
```

```
WHEN ODI_Unit='szt.' THEN 1
```

```
WHEN ODI_Unit='l' THEN 2
```

```
WHEN ODI_Unit='kg' THEN 3
```

```
Else 0
```

```
END, /*Jednostka miary*/
```

```
ODI_NetValue, /* Wartość Netto*/
```

CASE

WHEN ODI_VatRate='23.00' THEN 1

WHEN ODI_VatRate='8.00' THEN 2

WHEN ODI_VatRate='0.00' THEN 3

WHEN ODI_VatRate='5.00' THEN 7

Else 0

END,

ODI_GrossValue, /*Wartość Brutto*/

[ODI_GrossValue]-[ODI_NetValue], -- kwota VAT

1

from [BPM_Przykłady].do.OC_Document_Items I /*Zmienić na właściwą nazwę bazy BPM*/

join [BPM_Przykłady].do.OC_Documents D on
D.OCD_Id=I.ODI_DocumentId /*Zmienić na właściwą nazwę bazy BPM*/

where ODI_DocumentId=@ID_OCR@

END

Ta sekcja jest wywoływana przy każdej modyfikacji danych w kontrolce „Elementy” i wymaga użycia ELSE po sekcji OCR:

ELSE

If @^InitSenderControlName@ ='Elementy'

Begin

/*Deklaracja Stawki VAT */

```
DECLARE @VAT AS DECIMAL(4, 2)
```

```
/* Pobranie stawki VAT z tabeli CDN.Nazwy na podstawie kolumny  
StawkaVAT w bieżącym wierszu */
```

```
SELECT @VAT = CAST(SUBSTRING(Naz_Nazwa1, 0, CHARINDEX(''  
'', Naz_Nazwa1)) AS DECIMAL(4, 2)) / 100
```

```
FROM CDN.Nazwy
```

```
WHERE Naz_GIDTyp = 624
```

```
AND Naz_GIDLp = (
```

```
    SELECT [StawkaVAT]
```

```
    FROM @Elementy@
```

```
    WHERE POS = @Elementy_Row@
```

```
)
```

```
/* JEŚLI ZMIENIONO KOLUMNY: 2 (Cena), 3 (Ilość), 6 (Stawka  
VAT) */
```

```
IF @Elementy_Column@=2 OR @Elementy_Column@=3 OR @Elementy_Col  
umn@=6
```

```
BEGIN
```

```
    /* PRZELICZANIE W PRZÓD (Cena/Ilość/Stawka VAT ->  
Netto/Brutto) */
```

```
    /*Wyliczenie i aktualizacja netto i brutto */
```

```
    UPDATE @Elementy@
```

```
    SET
```

```
        [Netto] = (Ilosc * Cena),
```

```
        [Brutto] = (Ilosc * Cena) * (@VAT + 1)
```

```
    WHERE POS = @Elementy_Row@
```

```
    /* !!! OBOWIĄZKOWE: Odświeżenie wiersza !!! */
```

```
    SELECT * FROM @Elementy@
```

```
END
```

```
/* JEŚLI ZMIENIONO KOLUMNĘ: 7 (Brutto) */
```

```
IF @Elementy_Column@=7
```

```
BEGIN
```

```
    /* PRZELICZANIE W TYŁ (Brutto -> Cena/Netto) */
```

```
    /* Aktualizacja ceny i netto (Ilość stała) */
```

```
    UPDATE @Elementy@
```

```
    SET
```

```

[Cena] = [Brutto] / (@VAT + 1), /*Obliczenie
Ceny Netto z Brutto*/
[Netto] = [Ilosc] * ([Brutto] / (@VAT + 1)) /*Oblicze
nie Wartości Netto*/
WHERE POS = @Elementy_Row@

/* !!! OBOWIĄZKOWE: Odświeżenie wiersza !!! */
SELECT * FROM @Elementy@

END
END

```

The screenshot displays the SAP control panel interface. On the left, a window titled "Wyrażenie kontrolki" (Control Expression) contains a complex SQL query. The query includes conditional logic for VAT rates and unit types, and a final SELECT statement. Below the query editor, a legend shows the control type "Kontrolki" and its components: "UserId" and "DocumentId". On the right, the main control panel is visible, showing the "Dane tabelaryczne" (Table Data) tab. The "Inicjowanie wartości" (Value Initialization) section is active, with a dropdown menu set to "Brak" (None). Other sections include "Kontrolki powiązane" (Linked Controls) and "Wzór na wartość" (Value Formula).

Zapytanie SQL w kontrolce Dane tabelaryczne na zakładce Inicjowanie wartości

- **Tryb pracy:** W kontrolce typu **Dane tabelaryczne**, na zakładce **Ograniczenie edycji** kolumnę „Netto” ustawiono w trybie „Do odczytu”.

Dane tabelaryczne									
	OGÓLNE	LISTY	INICJOWANIE WARTOŚCI	OGRANICZENIE EDYCJI					
Zaznaczenie opcji spowoduje zablokowanie funkcjonalności lub edycji poniższych parametrów.									
Etap	<input type="checkbox"/> Zaznacz wszystko Zablokuj dodanie wierszy	<input type="checkbox"/> Zaznacz wszystko Zablokuj usuwanie wierszy	<input type="checkbox"/> Zaznacz wszystko Nazwa Nazwa	<input type="checkbox"/> Zaznacz wszystko Cena Cena	<input type="checkbox"/> Zaznacz wszystko Ilość Ilość	<input type="checkbox"/> Zaznacz wszystko JM Jednostka miary	<input checked="" type="checkbox"/> Zaznacz wszystko Netto Netto	<input type="checkbox"/> Zaznacz wszystko Stawka/VAT Stawka VAT	<input type="checkbox"/> Zaznacz wszystko Brutto Brutto
Rejestracja dokumentu	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Dekretacja kosztów	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Akceptacja kierownika	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Księgowość	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Odrzucono	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Archiwum	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Kolumna Netto w trybie „Do odczytu”

- **Utworzono kontrolkę liczbową:** Dodano kontrolkę typu **Liczba stałoprzecinkowa** o nazwie (identyfikatorze) **WartoscNetto**, w której dodano powiązanie do kontrolki **Elementy** i skonfigurowano w niej zapytanie, które zlicza wartość z kolumny **Netto** z kontrolki tabelarycznej:

```
select sum (Netto) from @Elementy@
```

Wyrażenie kontrolki

Uwaga!
Zmiana nazwy kontrolki może spowodować nieprawidłowe działanie wyrażenia.

```
1 SELECT SUM(Netto) FROM @Elementy@
```

Kontrolki

- ^Document
- ^UserId
- ^Document1
- ^Workflow
- ^Stageld

Składnia:
SELECT decimal (22,4)
decimal – wartość stałoprzecinkowa.
Przykład:
BEGIN TRY
SELECT 1000.2588
END TRY
BEGIN CATCH
--

Zapisz Zamknij

[Liczba stałoprzecinkowa] WartoscNetto

Liczba stałoprzecinkowa

ⓘ Kontrolkę można uzupełnić liczbą złożoną z maksymalnie 18 cyfr przed przecinkiem i 6 miejsc po przecink

Nazwa wyświetlana:

Nazwa (identyfikator):

Liczba miejsc po przecinku, do której będzie zaokrąglana wartość:

Prezentuj na całej szerokości:

Ustaw kontrolkę na początku wiersza:

Pokaż na liście:

Inicjowanie wartości

Inicjowanie:

Dokument:

Pole:

Zmiana wartości (Obserwator)

Kontrolki powiązane:

[Dodaj](#)

Wzór na wartość:

[Pokaż](#) [Usuń](#)

Kontrolka Liczbowa WartoscNetto

- **Utworzono kontrolkę liczbową:** Dodano kontrolkę typu **Liczba stałoprzecinkowa** o nazwie (identyfikatorze) **WartoscBrutto**, w której dodano powiązanie do kontrolki **Elementy** i skonfigurowano w niej zapytanie, które zlicza wartość z kolumny Brutto z kontrolki tabelarycznej:

```
select sum (Brutto) from @Elementy@
```

Wyrażenie kontrolki

Uwaga!
Zmiana nazwy kontrolki może spowodować nieprawidłowe działanie wyrażenia.

```
1 SELECT SUM(Brutto) FROM @Elementy@
```

Kontrolki

- ^Document
- ^UserId
- ^DocumentI
- ^Workflowc
- ^Stageld

Składnia:
SELECT decimal (22,4)
decimal – wartość stałoprzecinkowa.
Przykład:
BEGIN TRY
SELECT 1000.2588
END TRY
BEGIN CATCH
SELECT 0

Zapisz Zamknij

[Liczba stałoprzecinkowa] WartoscBrutto

Liczba stałoprzecinkowa

i Kontrolkę można uzupełnić liczbą złożoną z maksymalnie 18 cyfr przed przecinkiem i 6 miejsc po przecinku.

Nazwa wyświetlana:

Nazwa (identyfikator):

Liczba miejsc po przecinku, do której będzie zaokrąglana wartość:

Prezentuj na całej szerokości:

Ustaw kontrolkę na początku wiersza:

Pokaż na liście:

Inicjowanie wartości

Inicjowanie:

Dokument:

Pole:

Zmiana wartości (Obserwator)

Kontrolki powiązane: Elementy (Elementy)

[Dodaj](#)

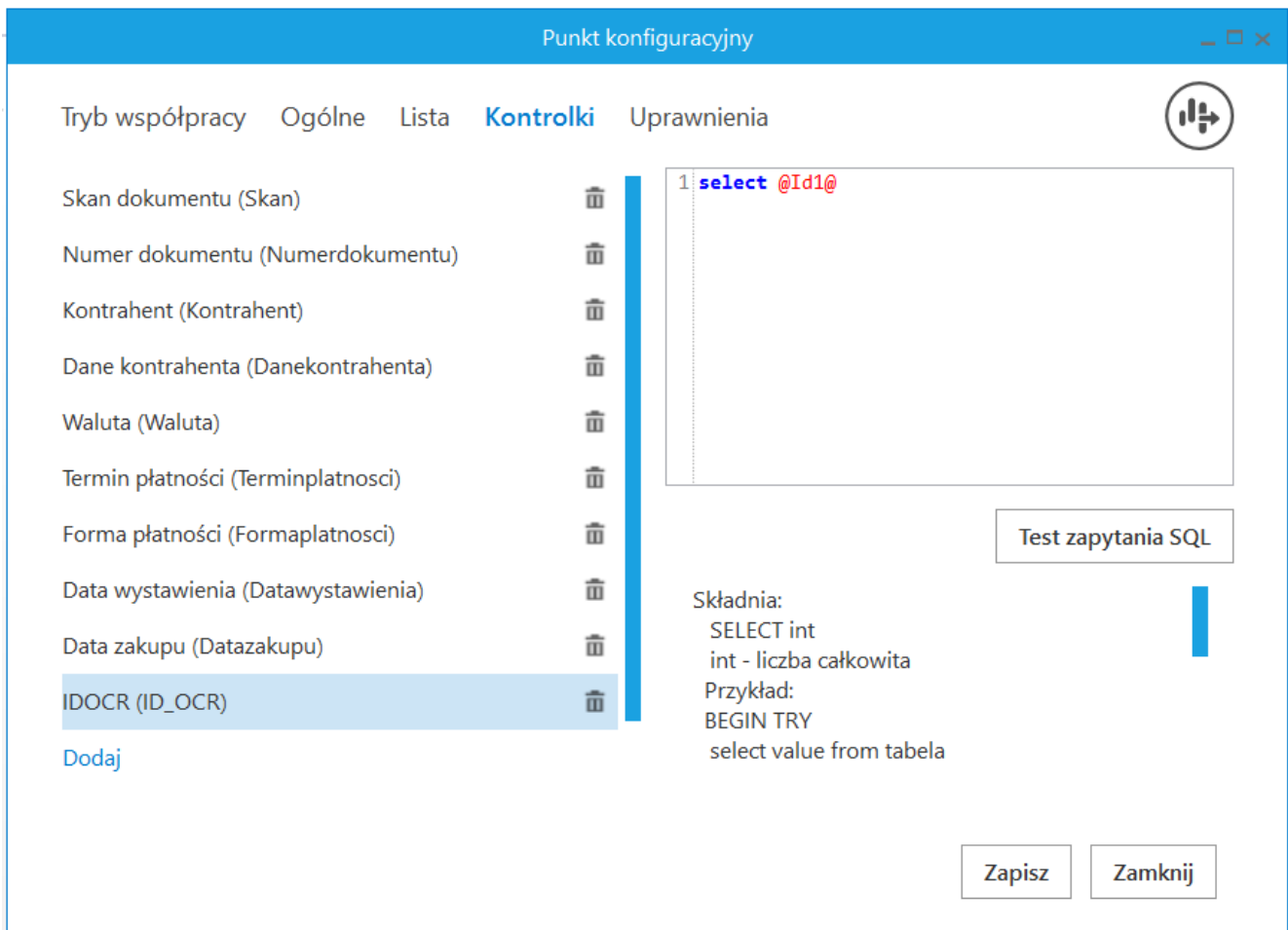
Wzór na wartość:

[Pokaż](#) [Usuń](#)

Kontrolka Liczbowa WartoscBrutto

- **Punkt ACD:** W punkcie konfiguracyjnym ACD (Współpraca z Comarch OCR), wykorzystywanym do inicjowania obiegu, na zakładce **Kontrolki** dodano kontrolkę **ID_OCR** i przypisano jej odpowiednie zapytanie:

```
select @Id1@
```



Kontrolka ID_OCR w punkcie ACD

Uzupełniona Kontrolka Dane Tabełaryczne

Po pomyślnym zainicjowaniu dokumentu z punktu ACD, kontrolka typu **Dane Tabełaryczne** została automatycznie wypełniona **pozycjami odczytanymi** przez **OCR**. Dla każdej pozycji, system automatycznie przeliczył i uzupełnił wartości w kolumnach **Netto** i **Brutto** w tym samym wierszu. W przypadku nieprawidłowości w kwotach możliwe jest przeliczanie wartości **w przód (Cena/Ilość -> Netto/Brutto)** oraz **w tył (Brutto -> Cena/Netto)**. Kolumna **Netto** jest zablokowana do ręcznej edycji (tryb „Do odczytu”), co gwarantuje poprawność danych. Kontrolka **WartoscNetto** wyświetla poprawną sumę końcową z kolumny **Netto** kontrolki Dane tabelaryczne. Kontrolka **WartoscBrutto** wyświetla poprawną sumę końcową z kolumny **Brutto**

kontrolki typu Dane tabelaryczne.

Pozycje dokumentu

Elementy

LP	Nazwa	Cena	Ilość	Jednostka miary	Netto	Stawka VAT	Brutto
1	Towar1	80,00	4,00	szt.	320,00	23%	393,60
2	Towar2	80,00	8,00	szt.	640,00	23%	787,20
3	Towar3	80,00	7,00	szt.	560,00	23%	688,80
4	Towar4	90,00	8,00	szt.	720,00	8%	777,60

Podsumowanie

WartoscNetto: 2 240,00 WartoscBrutto: 2 647,20

Techniczne

ID_OCR: 5

Kontrolka Dane tabelaryczne uzupełniona pozycjami odczytanymi przez Comarch OCR oraz podsumowanie kwot

Rozpoczynasz pracę z Comarch BPM (dawniej DMS) i chcesz dowiedzieć się, jak korzystać z programu? A może masz już podstawową wiedzę o Comarch BPM (dawniej DMS) i chcesz dowiedzieć się więcej?

[Sprawdź Szkolenia Comarch BPM!](#)

[Powrót do początku artykułu](#)

Czy wiesz, że można utworzyć nowego kontrahenta Comarch ERP XL z poziomu Comarch BPM?

Zastosowanie

Dzięki tej konfiguracji system Comarch BPM zyskuje możliwość **tworzenia nowych kontrahentów Comarch ERP XL** bezpośrednio z poziomu dokumentu w obiegu, po uzupełnieniu niezbędnych danych na formularzu bez potrzeby logowania się do Comarch ERPXL. Wdrożenie tego wzorca znacząco **przyspiesza** rejestrację takiego dokumentu oraz minimalizuje ryzyko podwójnego wprowadzania tych samych informacji.

Zawartość przykładu

- **Definicja Typu Obiegu (Workflow):** Wzorcowa ścieżka obiegu z prostym formularzem do uzupełnienia, kontrolką Własna akcja, która wywołuje procedurę **CDN.KntNowy** i zakładania kontrahenta.

[Kliknij tutaj, aby pobrać przykład](#)

Zasada działania

Mechanizm wykorzystuje kontrolkę typu **Własna akcja** (Dodaj

kontrahenta), która wywołuje standardową procedurę ERP XL – **cdn.KntNowy** bezpośrednio z obiegu dokumentów w BPM, eliminując konieczność ręcznego tworzenia kontrahenta w systemie ERP.

- Użytkownik uzupełnia pola obowiązkowe na formularzu BPM: **Operator zakładający w XL**, **Grupa kontrahentów**, **NIP_Prefiks** oraz **Akronim kontrahenta**. Opcjonalnie można uzupełnić dane adresowe i pełną nazwę oraz NIP.
- Parametry techniczne niezbędne do procedury ERP XL (takie jak **knwGrpTyp**, **knwGrpFirma**, **OpeTyp**, **OpeFirma**) wypełniają się automatycznie po dokonaniu wyboru w kontrolkach **Grupa kontrahentów** oraz **Operator zakładający w XL**. Pola te są jednocześnie oznaczone jako „**Tylko do odczytu**”, co zabezpiecza je przed przypadkową edycją z poziomu dokumentu.
- Po kliknięciu przycisku „**Dodaj kontrahenta**” (kontrolka typu **Własna akcja**), następuje wywołanie procedury **cdn.kntNowy**, przekazującej wszystkie zebrane dane do bazy ERP XL.
- Wywołanie to jest poprzedzone warunkiem (np. **IF @ERPContractor1_Id@ IS NULL**), aby zapobiec ponownemu utworzeniu kontrahenta i uniknąć błędu kontekstu SQL, który często pojawia się przy próbie wykonania **ROLLBACK** w procedurze.
- Następuje dodanie nowego kontrahenta w Comarch ERP XL.
- Na końcu, kontrolka typu **Kontrahent** na karcie obiegu uzupełnia się automatycznie numerem **GIDNumer** nowo utworzonego kontrahenta, na podstawie wprowadzonego akronimu.

Konfiguracja przykładu

Konfiguracja obejmuje zdefiniowanie zestawu kontrolek na karcie obiegu, z których część jest **widoczna** dla użytkownika, a część pełni rolę **ukrytych pól technicznych**, niezbędnych do poprawnego wywołania procedury ERP XL (**cdn.kntNowy**).

1. Konfiguracja Kontrolki Użytkownika i Własnej Akcji

Kontrolki te są widoczne na formularzu i służą do wprowadzenia kluczowych informacji o nowym kontrahencie.

- „Operator zakładający w XL” (identyfikator: Operator) – typ Lista
- „Grupa kontrahentów” (identyfikator: Grupa) – typ Lista
- „NIP_Prefiks” (identyfikator: NIP_Prefiks) – typ Lista
- „Akronim kontrahenta” (identyfikator: Akronim) – typ Tekst
- „Nazwa kontrahenta” (identyfikator: NazwaKontrahenta) – typ Tekst
- „NipE” (identyfikator: NipE) – typ Tekst
- „Kontrahent” (identyfikator: Kontrahent) – typ Kontrahent
- „Ulica i numer domu” (identyfikator: Ulica_Numer_Domu) – typ Tekst
- „Miasto” (identyfikator: Miasto) – typ Tekst
- „Kod pocztowy” (identyfikator: KodPocztowy) – typ Tekst
- „Dodaj kontrahenta XL” (identyfikator: DodajkontrahentaXL) – typ Własna akcja

2. Konfiguracja Ukrytych Kontrolki Technicznych

Pola te są niezbędne do przekazania GID-ów i typów do procedury `cdn.kntNowy`, wypełniają się automatycznie i są zablokowane do edycji:

- „knwGrpTyp” (identyfikator: knwGrpTyp) – typ Liczba całkowita
- „knwGrpFirma” (identyfikator: knwGrpFirma) – typ Liczba całkowita
- „knwGrpNumer” (identyfikator: knwGrpNumer) – typ Liczba całkowita

- „OpeTyp” (identyfikator: OpeTyp) – typ Liczba całkowita
- „OpeFirma” (identyfikator: OpeFirma) – typ Liczba całkowita
- „OpeNumer” (identyfikator: OpeNumer) – typ Liczba całkowita

KARTA OBIEGU SCHEMAT OBIEGU USTAWIENIA OBIEGU

Dane kontrahenta

Operator zakładający w XL

Grupa kontrahentów

Nazwa kontrahenta

Akronim kontrahenta

NIP_Prefiks

NipE

Kod podctowy

Miasto

Ulica i numer domu

Kontrahent [JEDNORAZOWY]

Parametry obowiązkowe- same się uzupełniają po wybraniu Grupy kontrahentów oraz Operatora

knwGrpTyp

knwGrpFirma

knwGrpNumer

OpeTyp

OpeFirma

OpeNumer

Formularz karty obiegu

3.Konfiguracja Inicjowania Wartości Pól Kontroltek (Pobieranie List)

- Operator zakładający w XL:
 - Typ Operacji: SQL OD

- **Zapytanie SQL:** Zapytanie warunkowo pobiera listę operatorów z bazy ERP XL. Jeśli zalogowany użytkownik (@^UserId@) ma powiązanie z operatorem ERP XL, domyślnie wybierany jest ten operator; w przeciwnym razie wyświetlana jest pełna lista.

```
IF EXISTS (SELECT * FROM cdn.OpeKarty /* baza ERP XL*/

JOIN      [BPM_Przykłady].do.DF_Conf0SDictionary      on
Ope_Ident=DCD_Login /*wsakzać właściwą nazwę bazy BPM*/

WHERE

DCD_ID= @^UserId@ AND

DCD_LoginMethod=0)

SELECT Ope_GIDNumer, Ope_Ident, Ope_GIDNumer

FROM cdn.OpeKarty /* baza ERP XL*/

JOIN      [BPM_Przykłady].do.DF_Conf0SDictionary      on
Ope_Ident=DCD_Login /*wsakzać właściwą nazwę bazy BPM*/

WHERE

DCD_ID= @^UserId@ AND

DCD_LoginMethod=0

ELSE

SELECT Ope_GIDNumer, Ope_Ident

FROM cdn.OpeKarty /* baza ERP XL*/

JOIN      [BPM_Przykłady].do.DF_Conf0SDictionary      on
Ope_Ident=DCD_Login /*wsakzać właściwą nazwę bazy BPM*/
```

WHERE

DCD_LoginMethod=0

Uwaga

W zapytaniu pobierającym listę operatorów dla kontrolki **Operator zakładający w XL**, konieczne jest **wstawienie właściwej nazwy bazy danych Comarch BPM** przed nazwą tabeli systemowej **do.DF_ConfOSDictionary**.

- **Grupa kontrahentów:**

- **Typ Operacji:** SQL ERP

- **Zapytanie SQL:** Pobiera listę wzorców grup kontrahentów z tabel **cdn.KntWzorce** i **cdn.KntGrupy**.

```
SELECT KnW_GrpNumer, KnG_Akronim FROM cdn.KntWzorce
```

```
JOIN cdn.KntGrupy ON kng_gidTyp=KnW_GrpTyp AND  
kng_gidnumer=KnW_GrpNumer where KnW_Nieaktywny=0
```

- **NIP_Prefiks:**

- **Typ Operacji:** SQL ERP

- **Zapytanie SQL:** Pobiera listę prefiksów NIP (kodów krajów) z tabeli **cdn.KrajeCelne**

```
select kpc_id, KPC_Kod, 168 FROM cdn.KrajeCelne
```

4. Konfiguracja Automatycznego Wypełniania Pól (Obserwator)

Poniższe reguły określają, jak pola techniczne są automatycznie wypełniane po wyborze wartości w kontrolkach

widocznych dla użytkownika.

- **knwGrpTyp**

- **Obserwuje:** Grupa kontrahentów
- **Typ Operacji:** SQL ERP
- **Zapytanie SQL:** Pobiera typ wzorca grupy z cdn.KntWzorce

```
SELECT CAST(Knw_GrpTyp as int) as int FROM cdn.KntWzorce WHERE Knw_grpnumer= @Grupakontrahentow_ID@
```

- **knwGrpFirma**

- **Obserwuje:** Grupa kontrahentów
- **Typ Operacji:** SQL ERP
- **Zapytanie SQL:** Pobiera firmę wzorca grupy z cdn.KntWzorce

```
SELECT Knw_GrpFirma FROM cdn.KntWzorce WHERE Knw_grpnumer= @Grupakontrahentow_ID@
```

- **knwGrpNumer**

- **Obserwuje:** Grupa kontrahentów
- **Typ Operacji:** SQL ERP
- **Zapytanie SQL:** Przekazuje numer wybranej grupy kontrahentów

```
SELECT @Grupakontrahentow_ID@
```

- **OpeTyp**

- **Obserwuje:** Operator zakładający w XL

- **Typ Operacji:** SQL ERP
- **Zapytanie SQL:** Pobiera GIDTyp operatora z cdn.OpeKarty

```
SELECT CAST (Ope_GIDTyp as int) as int FROM cdn.OpeKarty WHERE Ope_GIDNumer= @Operator_ID@
```

▪ OpeFirma

- **Obserwuje:** Operator zakładający w XL
- **Typ Operacji:** SQL ERP
- **Zapytanie SQL:** Pobiera GIDFirmę operatora z cdn.OpeKarty

```
SELECT Ope_GIDFirma FROM cdn.OpeKarty WHERE Ope_GIDNumer= @Operator_ID@
```

▪ OpeNumer

- **Obserwuje:** Operator zakładający w XL
- **Typ Operacji:** SQL ERP
- **Zapytanie SQL:** Przekazuje GIDNumer wybranego operatora

```
SELECT @Operator_ID@
```

▪ Kontrahent

- **Obserwuje:** Akronim kontrahenta, Dodaj kontrahenta
- **Typ Operacji:** SQL ERP

Zapytanie SQL: Pobiera GIDNumer nowego kontrahenta z

cdn.kntKarty na podstawie Akronimu.

```
IF @^SenderControlName@ ='DodajkontrahentaXL'
```

```
BEGIN
```

```
SELECT Knt_GIDNumer FROM cdn.kntKarty
```

```
WHERE knt_akronim=@Akronim@
```

```
END
```

```
ELSE
```

```
SELECT NULL
```

5. Konfiguracja Akcji Tworzenia Kontrahenta – Wywołuje procedurę ERP XL **cdn.kntNowy**, przekazując parametry zebrane z widocznych i ukrytych kontroltek:

- **Nazwa Kontrolki: Dodaj kontrahenta**
 - **Typ Operacji:** Procedura ERP
 - **Zapytanie SQL:** If @Kontrahent_Id@ is null

```
Begin
```

```
EXEC cdn.kntNowy
```

```
@knwGrpTyp=@knwGrpTyp@,
```

```
@knwGrpFirma=@knwGrpFirma@,
```

```
@knwGrpNumer=@knwGrpNumer@,
```

```
@OpeTyp=@OpeTyp@,
```

@OpeFirma=@OpeFirma@,

@OpeNumer=@OpeNumer@,

@Akronim=@Akronim@,

@Miasto= @Miasto@,

@NipPrefiks= @NIP_Prefiks_Text@ ,

@Nazwa1=@NazwaKontrahenta@,

@NipE=@NipE@,

@Ulica= @Ulica_Numer_Domu@ ,

@KodP= @KodPocztowy@

END

Kontrolki techniczne, które są uzupełniane automatycznie, zostały celowo oznaczone jako **Tylko do odczytu**, aby uniemożliwić ręczną edycję i zapewnić integralność danych przekazywanych do procedury ERP XL.

Etap

OGÓLNE UPRAWNIENIA PRZEKAZANIE MAIL **KONTROLKI** PARAMETRY

Włącz funkcję obserwatora podczas otwierania dokumentu

Kontrolka	<input checked="" type="checkbox"/> Zaznacz wszystko Widoczna	<input type="checkbox"/> Zaznacz wszystko Tylko do odczytu	<input type="checkbox"/> Zaznacz wszystko Wymagana	<input checked="" type="checkbox"/> Zaznacz wszystko Inicjalizacja Sql/C#	<input checked="" type="checkbox"/> Zaznacz wszystko Odświeżanie tabeli na żądanie	Kolejność inicjowania	
Zakładka 1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>			^
Dane kontrahenta	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>			v
Parametry obowiązkowe- sam	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>				^
knwGrpTyp	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>				v
knwGrpFirma	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>				
knwGrpNumer	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>				
OpeTyp	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>				
OpeFirma	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>				
OpeNumer	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>				

Ustawienie kontrolek Tylko do odczytu

Uwaga

Aby zablokować edycję kontrolek technicznych na **całym obiegu dokumentu**, należy pamiętać, że pole „**Tylko do odczytu**” musi zostać zaznaczone **na każdym etapie procesu**.

Uwaga

Jeżeli celem jest **ukrycie** kontrolek technicznych na dokumencie, należy pamiętać, że opcja „**Widoczna**” musi zostać **odznaczona na każdym etapie** obiegu dokumentu.

Etap

OGÓLNE UPRAWNIENIA PRZEKAZANIE MAIL **KONTROLKI** PARAMETRY

Włącz funkcję obserwatora podczas otwierania dokumentu

Kontrolka	<input type="checkbox"/> Zaznacz wszystko Widoczna	<input type="checkbox"/> Zaznacz wszystko Tylko do odczytu	<input type="checkbox"/> Zaznacz wszystko Wymagana	<input checked="" type="checkbox"/> Zaznacz wszystko Inicjalizacja Sql/C#	<input checked="" type="checkbox"/> Zaznacz wszystko Odświeżanie tabeli na żądanie	Kolejność inicjowania
Zakładka 1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		^
Dane kontrahenta	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		v
Parametry obowiązkowe- sam	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			^
knwGrpTyp	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			v
knwGrpFirma	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
knwGrpNumer	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
OpeTyp	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
OpeFirma	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
OpeNumer	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			

Ukrycie kontroltek technicznych

Utworzenie nowego kontrahenta za pomocą kontrolki Własna Akcja

Użytkownik uzupełnił niezbędne dane na karcie obiegu, w tym:

- Pola obowiązkowe: **Akronim kontrahenta**, **Operator zakładający w XL**, **Grupa kontrahentów** i **NIP_Prefiks**.
- Zostały również uzupełnione pola dodatkowe: dane adresowe, NIP i pełna nazwa kontrahenta.
- W momencie dokonania wyboru w kontrolkach **Operator zakładający w XL** i **Grupa kontrahentów**, system za pomocą reguł **Obserwatora (SQL ERP)** automatycznie pobrał identyfikatory GID i typy (np. knwGrpTyp, OpeFirma) z **ERP XL**. Pola te są zablokowane (**Tylko do odczytu**) na każdym etapie obiegu, co gwarantuje poprawność i integralność danych przekazywanych do procedury **ERP XL**.
- Po kliknięciu przycisku **„Dodaj kontrahenta”** (Własna akcja) została wywołana procedura ERP XL cdn.kntNowy, do której zostały przekazane zebrane dane i nastąpiło dodanie nowego kontrahenta w **Comarch ERP XL**.
- Kontrolka typu **Kontrahent** na formularzu uzupełniła się

automatycznie GIDNumerem nowo utworzonego kontrahenta, na podstawie wprowadzonego akronimu.

- Kliknięcie w kontrolkę **Kontrahent** otwiera widok nowego kontrahenta, **potwierdzając jego poprawne dodanie w Comarch ERP XL.**

Data utworzenia : 08-10-2025

Przebieg

Administrator
8.10.2025 16:01

Karta kontrahenta - zostanie zmieniona (TESTOWY1)

Ogólne Handlowe Limity kredytowe Rozliczenia Księgowe Kalendarze Inne Adresy Grupy i rejonny Zgody Osoby CRM

Akronim: TESTOWY1 NIP: PL 5678987654 Dostawca: Dostawca

Karta: Własna karta Begon: Pesel: Akwizytor: 0,00% Spedytor:

Typ: Kontrahent

Nazwa: TEST1 Telefon 1: Telefon 2: Fax: Modem: GSM: E-Mail: URL: Rejon:

Ulica: Słoneczna 3/7

Kraj: PL Kod: 20-560

Miasto: Kraków

Woj.: Powiat: Archiwalny: Adres nieaktualny: Wstrzymano transakcje: Wstrzymano zamówienia:

Gmina: GIN:

Typ: wartość

Dodaj

Osoby uprawnione do następnego etapu

Utworzenie nowego kontrahenta

Rozpoczynasz pracę z Comarch BPM (dawniej DMS) i chcesz dowiedzieć się, jak korzystać z programu? A może masz już podstawową wiedzę o Comarch BPM (dawniej DMS) i chcesz dowiedzieć się więcej?

[Sprawdź Szkolenia Comarch BPM!](#)

Czy wiesz, że po wybraniu z listy dokumentu PZ w kontrolce Dane tabelaryczne można pobrać do poszczególnych kolumn informacje z tego dokumentu?

Zastosowanie

Dzięki tej konfiguracji, proces obsługi **dokumentów w Comarch BPM** zyskuje zaawansowaną zdolność automatycznego pobierania danych z dokumentów **ERPXL**. Wdrożenie tego wzorca znacznie **przyspiesza procesy** zakupowe, ponieważ eliminuje konieczność ręcznego wyszukiwania i przepisywania danych z dokumentów (w przykładzie dokument Przyjęcia Zewnętrzne (PZ)). System automatycznie pobiera **kluczowe informacje** nagłówkowe (**Kontrahent, Wartość Netto**) z wybranego dokumentu PZ, uzupełniając **kontrolkę tabelaryczną** i wprowadzając **dane** bezpośrednio do obiegu.

Film obrazujący zastosowanie konfiguracji

<https://pomoc.comarch.pl/dms/wp-content/uploads/2025/06/pz2.mp4>

Zawartość przykładu

- **Definicja Typu Obiegu (Workflow):** Wzorcowa ścieżka obiegu dokumentu z konfiguracją kontrolki typu Dane tabelaryczne.

[Kliknij tutaj, aby pobrać przykład](#)

Zasada działania

Mechanizm opiera się na **obserwowaniu kolumny typu Lista** (lista wyboru dokumentu PZ) i uruchamianiu złożonego zapytania SQL, które **pobiera** powiązane **dane nagłówkowe** z tabel **Comarch ERP XL**.

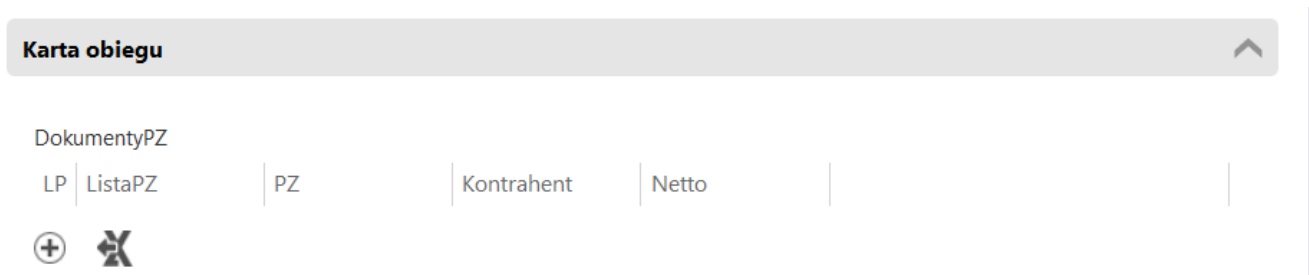
- W kontrolce typu **Dane Tabelaryczne** zdefiniowano kolumnę **ListaPZ** (typ: Lista) służącą do wyboru dokumentu **PZ** oraz kolumny wynikowe: **PZ** (typ: Dokument elektroniczny), **Kontrahent** (typ: Kontrahent) i **Netto** (typ: Liczba stałoprzecinkowa).
- Na zakładce **Listy** zdefiniowano zapytanie pobierające wszystkie **dokument PZ** (o typie Trn_GIdTyp=1489) z tabeli **cdn.TraNag**.
- Na zakładce **Inicjowanie wartości** dodano obserwację kontrolki przez samą siebie.
- Zapytanie **SQL** (Wzór na wartość SQL ERP) jest wyzwalane przy każdej zmianie kolumny **ListaPZ**.
- Zapytanie **SQL** pobiera **GIDNumer** wybranego dokumentu PZ i na jego podstawie odpytuje tabelę **cdn.TraNag** oraz **cdn.KntKarty** (dla Kontrahenta).
- Polecenie **UPDATE @DokumentyPZ@ SET** aktualizuje kolumny wynikowe w wierszu.

- Osobna kontrolka typu **SumaNettoPZ** obserwuje kolumnę (**Netto**) i zlicza jej sumę za pomocą prostego zapytania SQL (`select sum (Netto) from @DokumentyPZ@`).

Konfiguracja przykładu

1. **Utworzono kontrolkę Tabelaryczną:** Dodano kontrolkę typu **Dane tabelaryczne** o nazwie (identyfikatorze) **DokumentyPZ** i skonfigurowano w niej następujące kolumny:

- **"ListaPZ"** (identyfikator kolumny: **ListaPZ**) – typ **Lista**
- **"PZ"** (identyfikator kolumny: **PZ**) – typ **Dokument elektroniczny**
- **"Kontrahent"** (identyfikator kolumny: **Kontrahent**) – typ **Kontrahent**
- **"Netto"** (identyfikator kolumny: **Netto**) – typ **Liczba stałoprzecinkowa**



The screenshot shows a configuration window titled "Karta obiegu" with a table structure. The table has a header row "DokumentyPZ" and a data row with columns: "LP", "ListaPZ", "PZ", "Kontrahent", and "Netto". Below the table are two icons: a plus sign and a crossed-out X.

DokumentyPZ				
LP	ListaPZ	PZ	Kontrahent	Netto

Kolumny w kontrolce Dane tabelaryczne

2. W kontrolce typu **Dane tabelaryczne** na zakładce **Listy** zdefiniowano zapytanie pobierające:

- **Listę dokumentów PZ z Comarch ERPXL:**

```
select trn_gidnumer,cdn.numerdokumentu(trN_GIDTyp,TrN_SpiTyp,
Trn_TRNTyp,TrN_TrNNumer,TrN_TrNRok, TrN_TrNSeria, TrN_trNMiesi
ac) from cdn.TraNag where
Trn_GIdTyp=1489
```

[Dane tabelaryczne] DokumentyPZ

Dane tabelaryczne

OGÓLNE LISTY INICJOWANIE WARTOŚCI OGRANICZENIE EDYCJI

Lista 1

Nazwa: ListaPZ

Inicjowanie: SQL ERP

Rodzaj identyfikatora: Int

Tryb pracy: Lista

[Dodaj](#) [Pokaż](#) [Usuń](#)

Lista pobierająca dokumenty PZ w kontrolce typu Dane tabelaryczne

3. Dodano obserwację kontrolki: W kontrolce typu **Dane tabelaryczne**, na zakładce **Inicjowanie wartości**, dodano powiązanie do samej siebie.

Dane tabelaryczne

OGÓLNE

LISTY

INICJOWANIE WARTOŚCI

OGRANICZENIE EDYCJI

Inicjowanie wartości

Inicjowanie

Brak

Kontrolki powiązane

DokumentyPZ (DokumentyPZ)



Dodaj

Wzór na wartość

SQL ERP

Ustaw

Powiązanie kontrolki do samej siebie

4. Dodano zapytanie SQL: W kontrolce typu **Dane tabelaryczne**, na zakładce **Inicjowanie wartości** w sekcji **Wzór na wartość SQL ERP** dodano zapytanie, które automatycznie podpinia dokument elektroniczny w kolumnie **PZ**, oraz wczytuje kluczowe dane nagłówkowe (**Kontrahent** i **Netto**) bezpośrednio z wybranego dokumentu. Logika ta polega na trzykrotnym użyciu instrukcji **UPDATE @DokumentyPZ@ SET** w jednym bloku kodu. Każda instrukcja, oparta o połączenia (JOIN) z tabelami nagłówkowymi ERP XL (**cdn.TraNag**, **cdn.KntKarty**), pobiera i aktualizuje inną wartość w bieżącym wierszu tabeli.

```
IF @DokumentyPZ_Column@ = 1
```

BEGIN

/*[1] Zapisanie GID dokumentu PZ do kolumny PZ_ID w celu dalszych zapytań*/

UPDATE @DokumentyPZ@ SET [PZ_ID] =

(

SELECT [ListaPZ] FROM @DokumentyPZ@ WHERE POS = @DokumentyPZ_Row@

)

WHERE POS = @DokumentyPZ_Row@

/* [2] Pobranie Kontrahenta (Knt_GidNumer) z tabel nagłówkowych ERP XL*/

UPDATE @DokumentyPZ@ SET [Kontrahent] =

(

SELECT Knt_GidNumer FROM cdn.TraNag

JOIN cdn.kntkarty ON TrN_KntNumer = Knt_GIDNumer

WHERE TrN_GIDNumer = [ListaPZ]

)

WHERE POS = @DokumentyPZ_Row@

/*[3] Pobranie Wartości Netto (TrN_NettoP) z tabel TraNag/TraElem*/

UPDATE @DokumentyPZ@ SET [Netto] = (

SELECT TOP 1 TrN_NettoP

FROM cdn.TraNag

```
JOIN cdn.TraElem ON TrE_GIDNumer = TrN_GIDNumer
```

```
WHERE TrN_GIDNumer = [ListaPZ]
```

```
)
```

```
WHERE POS = @DokumentyPZ_Row@
```

```
/*!!! OBOWIĄZKOWE: Odświeżenie wiersza w kontrolce Dane tabelaryczne !!!*/
```

```
SELECT * FROM @DokumentyPZ@
```

```
END
```

The screenshot displays a software interface for configuring a control panel. On the left, a SQL query editor titled 'Wyrażenie kontrolki' contains the following code:

```
1 IF @DokumentyPZ_Column@ = 1
2 BEGIN
3 /*[1] Zapisanie GUID dokumentu PZ do kolumny PZ_ID w celu dalszych zapytań*/
4 UPDATE @DokumentyPZ@ SET [PZ_ID] =
5 (
6 SELECT [ListaPZ] FROM @DokumentyPZ@ WHERE POS = @DokumentyPZ_Row@
7 )
8 WHERE POS = @DokumentyPZ_Row@
9
10 /* [2] Pobranie Kontrahenta (Knt_GidNumer) z tabel nagłówekowych ERP XL*/
11 UPDATE @DokumentyPZ@ SET [Kontrahent] =
12 (
13 SELECT Knt_GidNumer FROM cdn.TraNag
14 JOIN cdn.kntkarty ON TrN_KntNumer = Knt_GIDNumer
15 WHERE TrN_GIDNumer = [ListaPZ]
16 )
17 WHERE POS = @DokumentyPZ_Row@
18
19 /*[3] Pobranie Wartości Netto (TrN_NettoP) z tabel TraNag/TraElem*/
20 UPDATE @DokumentyPZ@ SET [Netto] = (
21 SELECT TOP 1 TrN_NettoP
22 FROM cdn.TraNag
23 JOIN cdn.TraElem ON TrE_GIDNumer = TrN_GIDNumer
24 WHERE TrN_GIDNumer = [ListaPZ]
25 )
26 WHERE POS = @DokumentyPZ_Row@
27
28 /*!!! OBOWIĄZKOWE: Odświeżenie wiersza w kontrolce Dane tabelaryczne !!!*/
29 SELECT * FROM @DokumentyPZ@
30 END
31
```

Below the editor, a legend defines the variables used in the query:

Kontrolki	Składnia:
^UserId	SELECT
^DocumentId	int - Id Stanowiska (ListaPZ)
^WorkflowId	int, int - PZ_TYPE, PZ_ID
	int - ID (Kontrahent)
	decimal(22,4) - Netto

On the right, the 'Kontrolki' panel shows a search bar and a tabbed interface. The 'Dane tabelaryczne' tab is active, displaying a table with columns for 'Karta obiegu' and 'Suma Netto PZ'. Below the table, there are settings for 'Inicjowanie wartości' (set to 'Brak') and 'Wzór na wartość' (set to 'SQL ERP').

Zapytanie SQL w kontrolce typu Dane tabelaryczne na zakładce Inicjowanie wartości

5.Tryb pracy: W kontrolce typu Dane tabelaryczne, na zakładce

Ograniczenie edycji kolumnę „Kontrahent” oraz „Netto” ustawiono w trybie „Do odczytu”.

Dane tabelaryczne						
OGÓLNE	LISTY	INICJOWANIE WARTOŚCI	OGRANICZENIE EDYCJI			
Zaznaczenie opcji spowoduje zablokowanie funkcjonalności lub edycji poniższych parametrów.						
Etap	<input type="checkbox"/> Zaznacz wszystko Zablokuj dodanie wierszy	<input type="checkbox"/> Zaznacz wszystko Zablokuj usuwanie wierszy	<input type="checkbox"/> Zaznacz wszystko ListaPZ ListaPZ	<input type="checkbox"/> Zaznacz wszystko PZ PZ	<input checked="" type="checkbox"/> Zaznacz wszystko Kontrahent Kontrahent	<input checked="" type="checkbox"/> Zaznacz wszystko Netto Netto
Wprowadzenie	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Weryfikacja	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Archiwizacja	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Kolumna Netto w trybie „Do odczytu”

6. Utworzono kontrolkę liczbową: Dodano kontrolkę typu **Liczba stałoprzecinkowa** o nazwie (identyfikatorze) **SumaNettoPZ**, w której dodano powiązanie do kontrolki **DokumentyPZ** i skonfigurowano w niej zapytanie, które zlicza wartość z kolumny **Netto** z kontrolki tabelarycznej:

```
select sum (Netto) from @DokumentyPZ@
```

Wyrażenie kontrolki

[Liczba stałoprzecinkowa] Suma Netto PZ

Liczba stałoprzecinkowa

Uwaga!
Zmiana nazwy kontrolki może spowodować nieprawidłowe działanie wyrażenia.

```
1 select sum (Netto) from @DokumentyPZ@
```

Kontrolki

- ^Document
- ^Userid
- ^Document
- ^Workflowi
- ^Stageld

Składnia:
SELECT decimal
(22,4)
decimal – wartość
stałoprzecinkowa.
Przykład:
BEGIN TRY
SELECT 1000.2588
END TRY
BEGIN CATCH

Zapisz Zamknij

Informacja: Kontrolkę można uzupełnić liczbą złożoną z maksymalnie 18 cyfr przed przecinkiem i 6 miejsc po przecinku

Nazwa wyświetlana: SumaNettoPZ

Nazwa (identyfikator): SumaNettoPZ

Liczba miejsc po przecinku, do której będzie zaokrąglana wartość: 2

Prezentuj na całej szerokości:

Ustaw kontrolkę na początku wiersza:

Pokaż na liście:

Inicjowanie wartości

Inicjowanie: Brak

Dokument: [dropdown]

Pole: [dropdown]

Zmiana wartości (Obserwator)

Kontrolki powiązane: DokumentyPZ (DokumentyPZ) [trash icon]

Dodaj

Wzór na wartość: SQL OD

Pokaż Usun









Kontrolka Liczbowa Suma Netto PZ

Uzupełniona Kontrolka Dane Tabełaryczne

Po wybraniu z listy dokumentu **PZ** w kontrolce typu **Dane tabelaryczne**, wiersz został automatycznie zaktualizowany. W efekcie, w drugiej kolumnie podpiął się **dokument elektroniczny PZ** oraz zostały uzupełnione kluczowe dane nagłówkowe: **Kontrahent** i **Wartość Netto**. Kontrolka **SumaNettoPZ** wyświetla poprawną sumę końcową z kolumny **Netto** wszystkich podpiętych dokumentów **PZ**.

Karta obiegu

DokumentyPZ

LP	ListaPZ	PZ	Kontrahent	Netto
1	PZ-1/25	  PZ-1/25	  [COMARCH] COMARCH SPÓŁKA AKCYJNA	70,00
2	PZ-2/25	  PZ-2/25	  [ALOZA] Aloza S.A	80,00



Suma Netto PZ

150,00

Kontrolka typu Dane tabelaryczne z uzupełnionymi kolumnami

Dokument - Przyjęcie zewnętrzne [PZ-2/25] (rozliczony) - zostanie zmieniony

Ogólne Kontrahent Kaucje Nagłówek Księgowość VAT Intrastat Atrybuty Załączniki Do bufora

Kontrahent ALOZA Warszawa **Netto:** 80,00
 Aloza S.A. **Brutto:** 98,40
Kaucje: 0,00

Kraj wysyłki: PL Polska
 Transakcja: Krajowa
 Dostawa:
 Rodzaj transportu: 3 Transport drogowy

Płatność:

Lp.	Towar	Ilość	Jm.	Cena netto	Cena brutto	Wartości	
						Netto	Brutt
<input type="checkbox"/>	00001 TOWAR2	10,0000	szt.	8,0000 PLN	9,8400 PLN	80,00 PLN	98,40 PL

Filtr: MAG

Aktualizuj kaucje

Dokument PZ podpisany w kontrolce Dane tabelarycznej

Rozpoczynasz pracę z Comarch BPM (dawniej DMS) i chcesz dowiedzieć się, jak korzystać z programu? A może masz już podstawową wiedzę o Comarch BPM (dawniej DMS) i chcesz dowiedzieć się więcej?

[Sprawdź Szkolenia Comarch BPM!](#)

[Powrót do początku artykułu](#)