

Title: Wyszukanie kontrahentów w bazie

Subject: eDokumenty - elektroniczny system obiegu dokumentów, workflow i CRM -
DeployerGuide/Others/eDokumentyApi/SearchContacts

Version: 6

Date: 08/24/24 19:30:58

Table of Contents

Wyszukanie kontrahentów w bazie

3

Wyszukanie kontrahentów w bazie

Definicja parametrów:

```
<?php
/**
 * Szuka w bazie kontaktu/ów o podanych parametrach
 *
 * @param data Array - Tablica z parametrami w formie klucz => wartość, gdzie klucz to nazwa pola z bazy z tabeli contacts
 * natomiast wartość to ciąg tekstowy, który szukamy i który ma się znaleźć w kluczu
 *
 *
 * Możliwe jest określenie operatora logicznego, który zostanie użyty do łączenia wyrażeń/warunków.
 * Przykład:
 * ['(AND)name_2' => 'FIRMA', '(AND)city__' => 'warszawa',]
 * wyszuka kontakty których nazwa skrócona zawiera słowo "FIRMA" i miasto zawiera słowo "warszawa"
 * wygenerowany na tej podstawie SQL: (name_2 ~* 'FIRMA' AND city__ ~* 'warszawa')
 *
 * Możliwe jest również określenie operatora porównania (=, !=, ~, ~*) pola z podanym tekstem/wartością.
 * Przykład:
 * ['name_2=' => 'FIRMA']
 * wyszuka kontakty których nazwa skrócona jest równa "FIRMA" i miasto jest równe "Warszawa"
 * wygenerowany na tej podstawie SQL: (name_2 = 'FIRMA')
 *
 * @return Array - tablica identyfikatorów kontaktów - tablica może być pusta jeśli nie znalazł, SoapFault w razie niepowodzenia
 *
 * @throws Exception - SoapFault
 */
Array searchContacts(Array data)
?>
```

Przykłady wywołań:

```
// Plik MyService.php umieszczony w apps/edokumenty.
// MyService.php
<?php

define('EDOK_API_LOGIN', 'developer');
define('EDOK_API_PASSWORD', 'developer');
define('DEFAULT_ENTITY_SYMBOL', 'demo');

require_once('./classes/eDokumentyApi/EDokApiClient.inc');

$options = array(
    'location' => 'http://{host}:{port}/eDokumentyApi.php',
    'uri' => "eDokumentyAPI",
    'encoding'=>'UTF-8'
);

$client = new EDokApiClient(NULL, $options);
$client->setUser(EDOK_API_LOGIN);
$client->setPass(md5(EDOK_API_PASSWORD));
$header = new SoapHeader('eDokumentyAPI', 'entity_symbol', DEFAULT_ENTITY_SYMBOL);
$client->__setSoapHeaders($header);

$contact = array();
```

```
try {
    $data = array(
        'name_1' => 'SOAP TEST',
    );
    $contacts = $client->searchContacts($data);
    var_dump($contacts);
} catch(SoapFault $fault) {

    var_dump($fault);

    if ($fault->faultcode < 100) {
        trigger_error("SOAP Fault: (faultcode: {$fault->faultcode}, faultstring: {$fault->faultstring})", E_USER_ERROR);
    }
}

?>
```