

Dodaj komentarz

Definicja parametrów:

```
<?php

/**
 * Dodaje komentarz do wskazanego obiektu.
 *
 * @param array $data - Tablica z parametrami. Poniżej obsługiwane klucze.
 *     Int $data['prc_id'] - Identyfikator sprawy (processes.prc_id)
 *     Int $data['doc_id'] - Identyfikator dokumentu (documents.doc_id)
 *     Int $data['evntid'] - Identyfikator zdarzenia (events.evntid)
 *     Int $data['knbaid'] - Identyfikator dokumentu bazy wiedzy (knowledge_base.knbaid)
 *     String $data['commnt'] - Treść komentarza
 *     String $data['addat'] - Data dodania komentarza (nie jest wymagana w razie braku zostanie użyta bieżąca data)
 *     Int $data['adduid'] - Identyfikator pracownika dodającego komentarz (users.usr_id). Parametr nie jest wymagany i
 *     ma znaczenie tylko wtedy jeśli API uruchomiono z poziomu użytkownika EDOK_API. Jeśli API uruchomiono z poziomu uż
 *     Int $data['sendNotifies'] - 1 lub 0 czy ma wysyłać powiadomienie
 *
 * Parametry 'prc_id', 'doc_id', 'evntid', 'knbaid' są zamienne wystarczy podać 1. W przypadku podania kilku zostanie użyty
 *
 * @access public
 * @return Int -id komentarza jeśli sukces, 0 w razie niepowodzenia
 * @throws Exception
 * @since 2.5.8
 */
Int addComment(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);

$data = array(
    'doc_id' => 1,
    'commnt' => 'Ważna wiadomość',
    'adduid' => 666
```

```
);

try {
    $out = $client->addComment($data);
    var_dump($out);
} catch(SoapFault $fault) {
    var_dump($fault);

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

?>
```