

## Wysyłanie powiadomienia

### Definicja parametrów:

```
<?php

/**
 * Powiadom pracownika
 *
 * @param data Array - Tablica z parametrami
 * @param data['usr_id'] Int - identyfikator pracownika (users:usr_id)
 * @param data['msgtxt'] String - treść powiadomienia
 * @param data['type__'] Mixed - typ powiadomienia jako tablica lub tekst (kolejne typy oddzielone przecinkiem). Typy to:
 *
 * @param data['metaData'] Array - dodatkowe atrybuty do powiadomienia obsługiwane klucze to:
 * - ns.... to źródło powiadomień np. dokument
 * - nscls_ - klasa obiektu, którego dotyczy powiadomienie np. PROCESS, DOCUMENT
 * - nskey_ - identyfikator obiektu (klucz główny z właściwej tabeli np. processes, documents)
 * - nstxt_ - wyświetlany tekst
 *
 * - nt.... to target powiadomień np. sprawa czy dodanie dokumentu (source) do sprawy (target)
 * - ntcls_ - to samo co powyżej tylko dotyczy targetu
 * - ntkey_ - to samo co powyżej tylko dotyczy targetu
 * - nttxt_ - to samo co powyżej tylko dotyczy targetu
 * @return True - zawsze zwraca TRUE
 *
 * @throws Exception - SoapFault
 */
Int notifyUser(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(
    'usr_id' => 19,
    'msgtxt' => 'Ważna wiadomość'
```

```
);

try {
    $out = $client->notifyUser($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);
    }
}

?>
```