

Zwraca dane etapów danej procedury

Definicja parametrów:

```
<?php

/**
 * Zwraca dane etapów danej procedury.
 * Identyfikator procedury można pobrać wykonując $client->getProcess($prc_id) i w zwróconej
 * tablicy będzie kolumna procid.
 *
 * @param Int procid - Id procedury (procedures:procid)
 *
 * @return Array - tablica z danymi etapów jeśli pusta tzn, że procedura nie ma etapów
 *
 * @throws Exception - SoapFault
 */
Int getProceduresStages(Int procid)

?>
```

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);

$procid = 45;
$stages = array();

try {
    $stages = $client->getProceduresStages($procid);
    var_dump($stages);
} catch(SoapFault $fault) {
    var_dump($fault);

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

?>