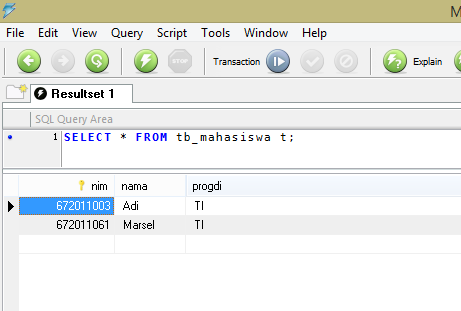
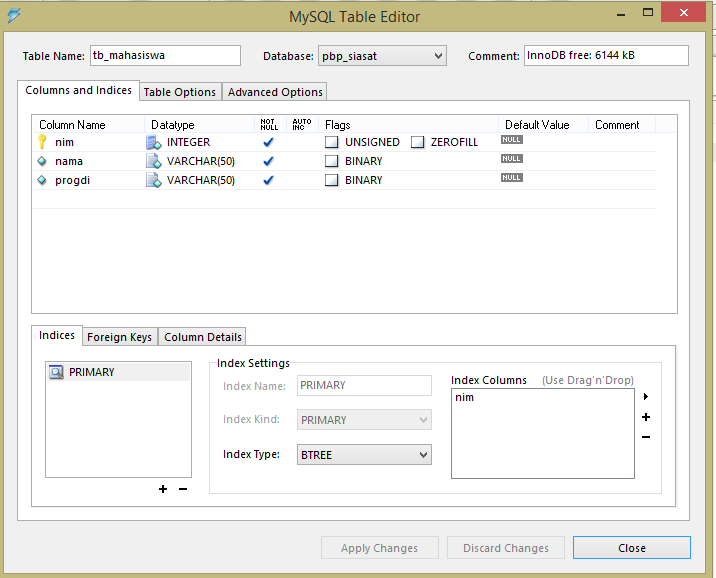
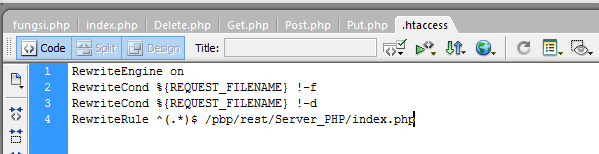
|  |  |
| --- | --- |
|  | **PEMROGRAMAN BERORIENTASI PLATFORM**  **REST**  Fakultas Teknologi Informasi  Universitas Kristen Satya Wacana  **Christine Dewi S.Kom, M.Cs**  **Marsel Sampe A. S.Kom** |

1. Buat database dengan struktur sebagai berikut : nama database = pbp\_siasat





1. Buat file pbp/rest/Server\_PHP/ .htaccess



1. pbp/rest/Server\_PHP/ index.php

<?php

$url\_array = explode('/', $\_SERVER['REQUEST\_URI']);

array\_shift($url\_array); // remove first value as it's empty // remove 2nd, 3rd and 4rd array, because it's directory

array\_shift($url\_array); // 2nd = 'pbp'

array\_shift($url\_array); // 3rd = 'rest'

array\_shift($url\_array); // 3rd = 'Server\_PHP'

// get the action (resource, collection)

$RESOURCE = $url\_array[0];

if($RESOURCE==""){

// jika tidak ada resource yang dituju

exit();

}

// get the method

$METHOD = $\_SERVER['REQUEST\_METHOD'];

include "fungsi.php";

$data=array();

if($RESOURCE=="mahasiswa"){

if($METHOD=="GET"){

if(!isset($url\_array[1])){ // if parameter nim not exist

$data = getAllMahasiswa();

}else{

$nim = $url\_array[1];

$data = getDetailMahasiswa($nim);

}

}elseif($METHOD=='POST'){

// get post from client

$jsonData = file\_get\_contents('php://input');

$mahasiswa = json\_decode($jsonData); // decode to object

$status = insertMahasiswa($mahasiswa);

$data = array('status'=>$status);

}elseif($METHOD=='PUT'){

// get put from client

$jsonData = file\_get\_contents('php://input');

$mahasiswa = json\_decode($jsonData); // decode to object

$nim = $url\_array[1];

$status = updateMahasiswa($nim, $mahasiswa);

$data = array('status'=>$status);

}elseif($METHOD=='DELETE'){

$nim = $url\_array[1];

$status = deleteMahasiswa($nim);

$data = array('status'=>$status);

}

}

// Set HTTP Response

header('HTTP/1.1 200 OK');

// Set HTTP Response Content Type

header('Content-Type: application/json; charset=utf-8');

// Print data

echo json\_encode($data);

?>

1. pbp/rest/Server\_PHP/ fungsi.php

<?php

// Create connection

function getDB() {

$servername = "127.0.0.1";

$username = "root";

$password = "";

$dbname = "pbp\_siasat";

$dbConnection = new PDO("mysql:host=$servername;dbname=$dbname", $username, $password);

$dbConnection->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

return $dbConnection;

}

function getAllMahasiswa()

{

$db = getDB();

try{

$result = array();

$query = "SELECT \* FROM tb\_mahasiswa";

$stmt = $db->prepare($query);

$stmt->execute();

$rs = $stmt->fetchAll(PDO::FETCH\_OBJ);

return $rs;

} catch(PDOException $e) {

return $e;

}

}

function getDetailMahasiswa($nim)

{

try{

$db = getDB();

$result = array();

$query = "SELECT \* FROM tb\_mahasiswa WHERE nim = :nim";

$stmt = $db->prepare($query);

$stmt->bindValue(':nim', $nim, PDO::PARAM\_STR);

$stmt->execute();

$rs = $stmt->fetch(PDO::FETCH\_OBJ);

return $rs;

} catch(PDOException $e) {

return $e;

}

}

function searchMahasiswa($nama)

{

$hasil = array();

try{

$db = getDB();

$result = array();

$query = "SELECT \* FROM tb\_mahasiswa WHERE nama LIKE :nama";

$stmt = $db->prepare($query);

$stmt->bindValue(':nama', '%'.$nama.'%', PDO::PARAM\_STR);

$stmt->execute();

$rs = $stmt->fetchAll(PDO::FETCH\_OBJ);

return $rs;

} catch(PDOException $e) {

return $e;

}

}

function insertMahasiswa($mhs){

try{

$db = getDB();

$query = "INSERT INTO tb\_mahasiswa VALUES (:nim, :nama, :progdi)";

$stmt = $db->prepare($query);

$stmt->bindValue(':nim', $mhs->nim, PDO::PARAM\_INT);

$stmt->bindValue(':nama', $mhs->nama, PDO::PARAM\_STR);

$stmt->bindValue(':progdi', $mhs->progdi, PDO::PARAM\_STR);

$stmt->execute();

return "success";

} catch(PDOException $e) {

return $e;

}

}

function updateMahasiswa($nimlama, $mhs){

try{

$db = getDB();

$query = "UPDATE tb\_mahasiswa SET nim=:nim, nama=:nama, progdi=:progdi WHERE nim=:nimlama";

$stmt = $db->prepare($query);

$stmt->bindValue(':nim', $mhs->nim, PDO::PARAM\_INT);

$stmt->bindValue(':nama', $mhs->nama, PDO::PARAM\_STR);

$stmt->bindValue(':progdi', $mhs->progdi, PDO::PARAM\_STR);

$stmt->bindValue(':nimlama', $nimlama, PDO::PARAM\_INT);

$stmt->execute();

return "success";

} catch(PDOException $e) {

return $e;

}

}

function deleteMahasiswa($nim)

{

try{

$db = getDB();

$query = "DELETE FROM tb\_mahasiswa WHERE nim=:nim";

$stmt = $db->prepare($query);

$stmt->bindValue(':nim', $nim, PDO::PARAM\_INT);

$stmt->execute();

return "success";

} catch(PDOException $e) {

return $e;

}

}

?>

1. pbp/rest/Client\_PHP /sampleClient/Get.php

<?php

// URL API

$url = 'http://localhost/pbp/rest/Server\_PHP/mahasiswa';

$client = curl\_init();

$options = array(

CURLOPT\_URL => $url, // Set URL of API

CURLOPT\_CUSTOMREQUEST => "GET", // Set request method

CURLOPT\_RETURNTRANSFER => true, // true, to return the transfer as a string

);

curl\_setopt\_array( $client, $options );

// Execute and Get the response

$response = curl\_exec($client);

// Get HTTP Code response

$httpCode = curl\_getinfo($client, CURLINFO\_HTTP\_CODE);

// Close cURL session

curl\_close($client);

// Show response

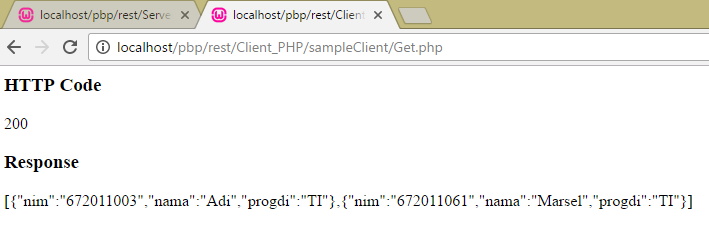
echo '<h3>HTTP Code</h3>';

echo $httpCode;

echo '<h3>Response</h3>';

echo $response;

?>



1. pbp/rest/Client\_PHP /sampleClient/Post.php

<?php

// URL API

$url = 'http://localhost/pbp/rest/Server\_PHP/mahasiswa';

$client = curl\_init();

$data = array("nim"=>"672006102", "nama"=>"Christine", "progdi"=>"TI");

$data = json\_encode($data);

$options = array(

CURLOPT\_URL => $url, // Set URL of API

CURLOPT\_CUSTOMREQUEST => "POST", // Set request method

CURLOPT\_RETURNTRANSFER => true, // true, to return the transfer as a string

CURLOPT\_POSTFIELDS => $data, // Send the data in HTTP POST

);

curl\_setopt\_array( $client, $options );

// Execute and Get the response

$response = curl\_exec($client);

// Get HTTP Code response

$httpCode = curl\_getinfo($client, CURLINFO\_HTTP\_CODE);

// Close cURL session

curl\_close($client);

// Show response

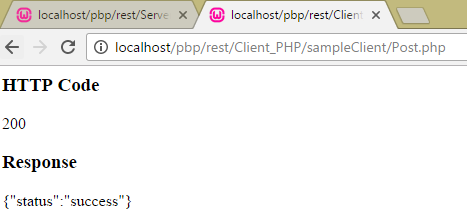
echo '<h3>HTTP Code</h3>';

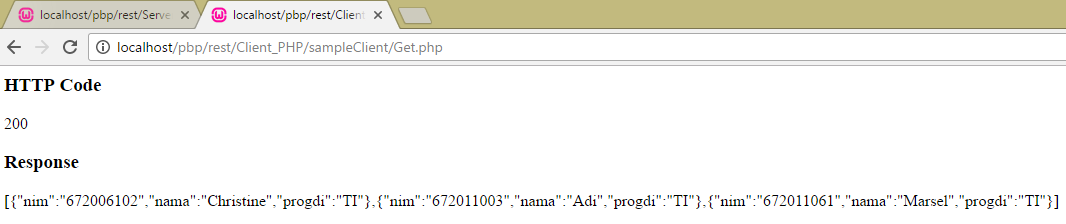
echo $httpCode;

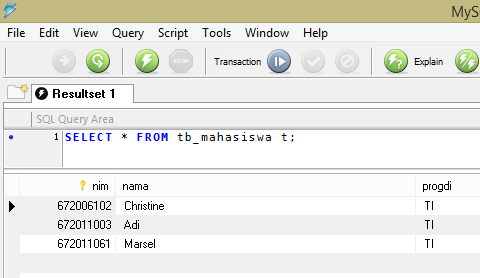
echo '<h3>Response</h3>';

echo $response;

?>







1. pbp/rest/Client\_PHP /sampleClient/Put.php

<?php

$nim = '672011003'; // id that will update

// URL API

$url = 'http://localhost/pbp/rest/Server\_PHP/mahasiswa/'.$nim;

$client = curl\_init();

$data = array("nim"=>"672011004", "nama"=>"Budi", "progdi"=>"TI");

$data = json\_encode($data);

$options = array(

CURLOPT\_URL => $url, // Set URL of API

CURLOPT\_CUSTOMREQUEST => "PUT", // Set request method

CURLOPT\_RETURNTRANSFER => true, // true, to return the transfer as a string

CURLOPT\_POSTFIELDS => $data, // Send the data in HTTP POST

);

curl\_setopt\_array( $client, $options );

// Execute and Get the response

$response = curl\_exec($client);

// Get HTTP Code response

$httpCode = curl\_getinfo($client, CURLINFO\_HTTP\_CODE);

// Close cURL session

curl\_close($client);

// Show response

echo '<h3>HTTP Code</h3>';

echo $httpCode;

echo '<h3>Response</h3>';

echo $response;

?>

1. pbp/rest/Client\_PHP /sampleClient/Delete.php

<?php

$nim = '672011001'; // id that will delete

// URL API

$url = 'http://localhost/pbp/rest/Server\_PHP/mahasiswa/'.$nim;

$client = curl\_init();

$options = array(

CURLOPT\_URL => $url, // Set URL of API

CURLOPT\_CUSTOMREQUEST => "DELETE", // Set request method

CURLOPT\_RETURNTRANSFER => true, // true, to return the transfer as a string

);

curl\_setopt\_array( $client, $options );

// Execute and Get the response

$response = curl\_exec($client);

// Get HTTP Code response

$httpCode = curl\_getinfo($client, CURLINFO\_HTTP\_CODE);

// Close cURL session

curl\_close($client);

// Show response

echo '<h3>HTTP Code</h3>';

echo $httpCode;

echo '<h3>Response</h3>';

echo $response;

?>

1. pbp/rest/Client\_PHP /sampleClient/Search.php (by nim)

<?php

$nim = '672006102'; // id that will search

// URL API

$url = 'http://localhost/pbp/rest/Server\_PHP/mahasiswa/'.$nim;

$client = curl\_init();

$options = array(

CURLOPT\_URL => $url, // Set URL of API

CURLOPT\_CUSTOMREQUEST => "GET", // Set request method

CURLOPT\_RETURNTRANSFER => true, // true, to return the transfer as a string

);

curl\_setopt\_array( $client, $options );

// Execute and Get the response

$response = curl\_exec($client);

// Get HTTP Code response

$httpCode = curl\_getinfo($client, CURLINFO\_HTTP\_CODE);

// Close cURL session

curl\_close($client);

// Show response

echo '<h3>HTTP Code</h3>';

echo $httpCode;

echo '<h3>Response</h3>';

echo $response;

?>



TUGAS

2. Buatlah tampilan untuk CRUD REST sebagai berikut :

