|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | **set up the two CodeIgniter projects, where one exposes data via a REST API and the other consumes it** | | |
| **Step:** | | **Headings** | **Details** | |
| TS00.47 | | Download Code igniter | <https://codeigniter.com/userguide3/installation/downloads.html> | |
| TS 00.48 | | Copy paste in htdocs folder in Xampp | C:\xampp\htdocs\6\_api\funda\_api\_s1\application\controllers\Welcome.php  C:\xampp\htdocs\1\_api  http://localhost/1\_api/funda/ | |
|  | | 1. First Project (API Provider) |  | |
|  | | Download the Required file | 1. RestController.php 2. Rest\_controller\_lang.php 3. Rest.php 4. Format.php   <https://www.fundaofwebit.com/post/codeigniter-3-restful-api-tutorial-using-postman#google_vignette> | |
|  | | Paste the language file in language folder | C:\xampp\htdocs\6\_api\funda\_api\_s1\application\language\english\rest\_controller\_lang.php | |
|  | | Code are here |  | |
|  | | Copy the( rest.php ) and pestle the following path | C:\xampp\htdocs\6\_api\funda\_api\_s1\application\config\rest.php | |
|  | | Code are here |  | |
|  | | Copy the restContrer.php & Format.php int he following path | C:\xampp\htdocs\6\_api\funda\_api\_s1\application\libraries\RestController.php  C:\xampp\htdocs\6\_api\funda\_api\_s1\application\libraries\Format.php | |
|  | |  |  | |
|  | | `Composer check | Open >new terminal>  Write “Composer” | |
|  | | Call page through composer | php -S localhost:8000 | |
|  | | Call the page | <http://localhost:8000>  <http://localhost/1_api/API_Provider/> | |
|  | | C:\1_desgin\05-09-2024_Api&ci\15-09-2024\api\ds1.png | | |
|  | |  |  | |
|  | | Controller create :  C:\xampp\htdocs\1\_api\API\_Provider\application\controllers\ApiDemo.php | **1. First Project (API Provider)**  In your first project, you have the controller ApiDemoController. Let's refine it a bit by preparing to send data.  **ApiDemoController:**  php  Copy code  <?php  defined('BASEPATH') OR exit('No direct script access allowed');  require APPPATH . 'libraries/RestController.php';  require APPPATH . 'libraries/Format.php';  use chriskacerguis\RestServer\RestController;  class ApiDemo extends RestController  {      public function \_\_construct()      {          parent::\_\_construct();          // Load models if necessary      }      public function users\_get()      {          // Sample data that will be returned as JSON          $users = [              ['id' => 1, 'name' => 'John Doe', 'email' => 'john@example.com'],              ['id' => 2, 'name' => 'Jane Smith', 'email' => 'jane@example.com']          ];          // Respond with data and HTTP status code          $this->response($users, RestController::HTTP\_OK);      }  }  ?>   * **Explanation:**   + The users\_get function returns a list of users as JSON.   + You could modify this to fetch data from your database using models. | |
|  | | Route create | $route['api'] = 'ApiDemo/users';  C:\xampp\htdocs\1\_api\API\_Provider\application\config\routes.php | |
|  | |  |  | |
|  | | Call the page | http://localhost:8000/api | |
|  | |  |  | |
|  | |  |  | |
|  | | ****2. Second Project (API Consumer)**** In the second project, you need to make an HTTP request to the first project's API and handle the response. |  | |
|  | | ApiClientController: | C:\xampp\htdocs\1\_api\API\_Consumer\application\controllers\ApiClientController.php | |
|  | |  | <?php  defined('BASEPATH') OR exit('No direct script access allowed');  class ApiClientController extends CI\_Controller  {      public function \_\_construct()      {          parent::\_\_construct();      }      public function get\_users()      {          // The API URL of the first project          $api\_url = 'http://localhost:8000/api';          // Initialize cURL session          $ch = curl\_init();          // Set cURL options          curl\_setopt($ch, CURLOPT\_URL, $api\_url);          curl\_setopt($ch, CURLOPT\_RETURNTRANSFER, true);          // Execute the cURL request          $response = curl\_exec($ch);          // Check for errors          if ($response === false) {              echo "cURL Error: " . curl\_error($ch);          } else {              // Decode the response              $data['users'] = json\_decode($response, true);              // Load view and pass the data (You can replace 'users\_view' with your actual view)              $this->load->view('users\_view', $data);          }          // Close the cURL session          curl\_close($ch);      }  }  ?> | |
|  | | View to Display Data (users\_view.php) | C:\xampp\htdocs\1\_api\API\_Consumer\application\views\users\_view.php | |
|  | |  | <html>  <head>      <title>Users List</title>  </head>  <body>      <h1>List of Users</h1>      <ul>          <?php if (!empty($users)) : ?>              <?php foreach ($users as $user) : ?>                  <li><?php echo $user['name']; ?> (<?php echo $user['email']; ?>)</li>              <?php endforeach; ?>          <?php else : ?>              <li>No users found.</li>          <?php endif; ?>      </ul>  </body>  </html> | |
|  | | ****Second Project (Client Routes):**** In application/config/routes.php, add the route for the client: | $route['client/get\_users'] = 'ApiClientController/get\_users'; | |
|  | | **Run the composer** | **php -S localhost:8080**  [**http://localhost/1\_api/API\_Consumer/**](http://localhost/1_api/API_Consumer/) | |
|  | | Call the page | <http://localhost:8080/client/get_users>  http://localhost/1\_api/API\_Consumer/client/get\_users | |
|  | |  |  | |
|  | | 1. First Project (API Provider) |  | |
|  | |  | …or create a new repository on the command line echo "# API\_Consumer" >> README.md  git init  git add README.md  git commit -m "first commit"  git branch -M master  git remote add origin https://github.com/Code-with-nandu/API\_Consumer.git  git push -u origin master …or push an existing repository from the command line git remote add origin https://github.com/Code-with-nandu/API\_Consumer.git  git branch -M master  git push -u origin master | |
|  | | ****2. Second Project (API Consumer)**** |  | |
|  | |  | …or create a new repository on the command line echo "# API\_Provider" >> README.md  git init  git add README.md  git commit -m "first commit"  git branch -M master  git remote add origin https://github.com/Code-with-nandu/API\_Provider.git  git push -u origin master …or push an existing repository from the command line git remote add origin https://github.com/Code-with-nandu/API\_Provider.git  git branch -M master  git push -u origin master | |
|  | |  |  | |
|  | | 1. First Project (API Provider) | Call Data from database create : | |
|  | |  |  | |
|  | |  | git branch  git checkout -b krish/2nd  git branch | |
|  | | C:\xampp\htdocs\1\_api\API\_Provider\application\config\autoload.php | $autoload['libraries'] = array('database'); | |
|  | |  | $active\_group = 'default';  $query\_builder = TRUE;  $db['default'] = array(      'dsn'   => '',      'hostname' => 'localhost',      'username' => 'root',      'password' => '',      'database' => 'busbooking',      'dbdriver' => 'mysqli',      'dbprefix' => '',      'pconnect' => FALSE,      'db\_debug' => (ENVIRONMENT !== 'production'),      'cache\_on' => FALSE, | |
|  | | C:\xampp\htdocs\1\_api\API\_Provider\application\models\Employee\_model.php | <?php  defined('BASEPATH') OR exit('No direct script access allowed');  class Employee\_model extends CI\_Model {      // Method to insert data into the employee table      public function insert\_employee()      {          $query = $this->db->get('employee');          return  $query->result();      }  }  ?> | |
|  | | C:\xampp\htdocs\1\_api\API\_Provider\application\controllers\ApiDemo.php | <?php  defined('BASEPATH') OR exit('No direct script access allowed');  require APPPATH . 'libraries/RestController.php';  require APPPATH . 'libraries/Format.php';  use chriskacerguis\RestServer\RestController;  class ApiDemo extends RestController  {      public function \_\_construct()      {          parent::\_\_construct();          $this->load->model('Employee\_model');          // Load models if necessary      }      public function users\_get()      {          $emp = new Employee\_model;          $result\_emp = $emp ->insert\_employee();              // Respond with data and HTTP status code          $this->response($result\_emp, RestController::HTTP\_OK);      }  }  ?> | |
|  | |  | php -S localhost:8000 | |
|  | | Call data through postman | <http://localhost:8000/api>  <http://localhost/1_api/API_Provider/api> | |
|  | |  |  | |
|  | |  |  | |
|  | | ****2. Second Project (API Consumer)**** | C:\xampp\htdocs\1\_api\API\_Consumer\application\views\users\_view.php | |
|  | |  | <html>  <head>      <title>Users List</title>  </head>  <body>      <h1>List of Users</h1>      <ul>          <?php if (!empty($users)) : ?>              <?php foreach ($users as $user) : ?>                  <li><?php echo $user['id']; ?> (<?php echo $user['first\_name']; ?>)<?php echo $user['last\_name']; ?> (<?php echo $user['phone']; ?>)<?php echo $user['email']; ?></li>              <?php endforeach; ?>          <?php else : ?>              <li>No users found.</li>          <?php endif; ?>      </ul>  </body>  </html> | |
|  | | Run composer | **php -S localhost:8080** | |
|  | | Call the page | http://localhost:8080/client/get\_users | |
|  | |  |  | |
|  | | Display in table | <!DOCTYPE html>  <html>  <head>      <title>Users List</title>      <style>          table {              width: 100%;              border-collapse: collapse;          }          table, th, td {              border: 1px solid black;          }          th, td {              padding: 8px;              text-align: left;          }          th {              background-color: #f2f2f2;          }      </style>  </head>  <body>      <h1>List of Users</h1>      <?php if (!empty($users)) : ?>          <table>              <thead>                  <tr>                      <th>ID</th>                      <th>First Name</th>                      <th>Last Name</th>                      <th>Phone</th>                      <th>Email</th>                  </tr>              </thead>              <tbody>                  <?php foreach ($users as $user) : ?>                      <tr>                          <td><?php echo htmlspecialchars($user['id']); ?></td>                          <td><?php echo htmlspecialchars($user['first\_name']); ?></td>                          <td><?php echo htmlspecialchars($user['last\_name']); ?></td>                          <td><?php echo htmlspecialchars($user['phone']); ?></td>                          <td><?php echo htmlspecialchars($user['email']); ?></td>                      </tr>                  <?php endforeach; ?>              </tbody>          </table>      <?php else : ?>          <p>No users found.</p>      <?php endif; ?>  </body>  </html> | |
|  | | Call the page again | <http://localhost:8080/client/get_users> | |
|  | |  |  | |
|  | |  |  | |
|  | | 1. First Project (API Provider) | Inserting Data **Second Project (API Consumer)** | |
|  | | Controller function create for insering data  C:\xampp\htdocs\1\_api\API\_Provider\application\controllers\ApiDemo.php | public function storeEmp\_post(){          $emp = new Employee\_model;          $data=[              'first\_name'=> $this->input->post('first\_name'),              'last\_name'=> $this->input->post('last\_name'),              'phone'=> $this->input->post('phone'),              'email'=> $this->input->post('email'),          ];          $result =$emp ->inn\_employee($data);          // $this ->response($data,$resul,200);          if($result>0)          {              $this->response([                  'status'=> true,                  'message'=> 'New Employee created'              ] ,RestController::HTTP\_OK );          }          else          {              $this->response([                  'status'=> false,                  'message'=> 'Failed to  created employee'              ] ,RestController::HTTP\_BAD\_REQUEST );          }      } | |
|  | |  |  | |
|  | | Model function create  C:\xampp\htdocs\1\_api\funda\_2\application\models\Employee\_model.php | public function inn\_employee($data)      {          return $this ->db->insert('employee',$data);      } | |
|  | | Route Create  C:\xampp\htdocs\1\_api\API\_Provider\application\config\routes.php | $route['api/store'] = 'ApiDemo/storeEmp'; | |
|  | | Run the project | php -S localhost:8000 | |
|  | | First check data insertion through postman | <http://localhost:8000/api/store>  <http://localhost/1_api/API_Provider/api/store> | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | | **To insert data from your Second Project (API Consumer) into the First Project (API Provider) via API, you will need to create a controller that sends data using an HTTP POST request.** | | |
|  | | Inserting Data **Second Project (API Consumer)** |  | |
|  | |  | <?php  defined('BASEPATH') or exit('No direct script access allowed');  class ApiClientController extends CI\_Controller  {      public function \_\_construct()      {          parent::\_\_construct();          $this->load->library('curl'); // Load cURL library          $this->load->helper('url');          $this->load->library('session');      }      public function get\_users()      {          // The API URL of the first project          $api\_url = 'http://localhost:8000/api';          // Initialize cURL session          $ch = curl\_init();          // Set cURL options          curl\_setopt($ch, CURLOPT\_URL, $api\_url);          curl\_setopt($ch, CURLOPT\_RETURNTRANSFER, true);          // Execute the cURL request          $response = curl\_exec($ch);          // Check for errors          if ($response === false) {              echo "cURL Error: " . curl\_error($ch);          } else {              // Decode the response              $data['users'] = json\_decode($response, true);              // Load view and pass the data (You can replace 'users\_view' with your actual view)              $this->load->view('users\_view', $data);          }          // Close the cURL session          curl\_close($ch);      }      public function storeEmployee()      {          // Data to be sent to the API          $data = [              'first\_name' => $this->input->post('first\_name'),              'last\_name'  => $this->input->post('last\_name'),              'phone'      => $this->input->post('phone'),              'email'      => $this->input->post('email'),          ];          // cURL configuration for POST request to the API          $url = 'http://localhost:8000/api/store'; // API endpoint          $response = $this->curl->simple\_post($url, $data);          if ($response) {              $result = json\_decode($response);              if ($result->status) {                  $this->session->set\_flashdata('success', $result->message);              } else {                  $this->session->set\_flashdata('error', $result->message);              }          } else {              $this->session->set\_flashdata('error', 'Failed to communicate with the API');          }          // Redirect to the form page          redirect('client/get\_users');      }      public function form()      {          // Load a view for the form          $this->load->view('employee\_form');      }    }  ?> | |
|  | | View Page create :  C:\xampp\htdocs\1\_api\API\_Consumer\application\views\employee\_form.php | <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta http-equiv="X-UA-Compatible" content="IE=edge">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Employee Form</title>      <style>          /\* Set background image for the entire page dynamically using base\_url() \*/          body {              background-image: url('<?php echo base\_url('assets/image/images.jpg'); ?>');              background-size: cover;              background-position: center;              background-repeat: no-repeat;              font-family: Arial, sans-serif;              color: #fff; /\* Text color for visibility \*/          }          h1 {              text-align: center;              margin-top: 20px;              color: #ffffff;          }          form {              background-color: rgba(0, 0, 0, 0.6); /\* Add a darker semi-transparent background to the form \*/              padding: 30px;              margin: 50px auto;              width: 400px; /\* Increase the width of the form \*/              border-radius: 10px;              box-shadow: 0 0 10px rgba(0, 0, 0, 0.5); /\* Add shadow for depth \*/              color: #fff;          }          label, input {              display: block;              margin-bottom: 15px;              font-size: 1.2em; /\* Make the font size larger \*/          }          input[type="text"], input[type="email"] {              width: 100%;              padding: 10px;              font-size: 1.1em;              border-radius: 5px;              border: none;          }          input[type="submit"] {              width: 100%;              background-color: #4CAF50;              color: white;              padding: 12px;              font-size: 1.2em;              border: none;              border-radius: 5px;              cursor: pointer;          }          input[type="submit"]:hover {              background-color: #45a049;          }          p {              text-align: center;          }      </style>  </head>  <body>      <h1>Insert Employee Data</h1>        <!-- Display success or error message -->      <?php if($this->session->flashdata('success')): ?>          <p style="color:green;"><?php echo $this->session->flashdata('success'); ?></p>      <?php elseif($this->session->flashdata('error')): ?>          <p style="color:red;"><?php echo $this->session->flashdata('error'); ?></p>      <?php endif; ?>      <form action="<?php echo base\_url('ApiClientController/storeEmployee'); ?>" method="post">          <label for="first\_name">First Name:</label>          <input type="text" id="first\_name" name="first\_name" required>          <label for="last\_name">Last Name:</label>          <input type="text" id="last\_name" name="last\_name" required>          <label for="phone">Phone:</label>          <input type="text" id="phone" name="phone" required>          <label for="email">Email:</label>          <input type="email" id="email" name="email" required>          <input type="submit" value="Submit">      </form>  </body>  </html> | |
|  | | **Curl file**  **(add the file libraries)** | <https://github.com/philsturgeon/codeigniter-curl/blob/master/libraries/Curl.php> | |
|  | | Route create  C:\xampp\htdocs\1\_api\API\_Consumer\application\config\routes.php | $route['client/store'] = 'ApiClientController/form'; | |
|  | | Base Url Create  C:\xampp\htdocs\1\_api\API\_Consumer\application\config\config.php | $config['base\_url'] = 'http://localhost:8080/'; | |
|  | | Autoload file create  C:\xampp\htdocs\1\_api\API\_Consumer\application\config\autoload.php | $autoload['libraries'] = array('curl','session');  $autoload['helper'] = array('url'); | |
|  | |  |  | |
|  | |  |  | |
|  | |  | php -S localhost:8080 | |
|  | |  | <http://localhost:8000/api/store> | |
|  | |  |  | |
|  | | 1. First Project (API Provider) | Specific data call Data (**Find employee by ID)** | |
|  | |  |  | |
|  | | Route create :  C:\xampp\htdocs\1\_api\API\_Provider\application\config\routes.php | $route['api/find/(:any)'] ='ApiDemo/findEmp/$1'; | |
|  | | Model function create specific dat calling  C:\xampp\htdocs\1\_api\API\_Provider\application\models\Employee\_model.php | public function findEmp($id)      {          $this->db->where('id',$id);          $query = $this->db->get('employee');          return $query->row();      } | |
|  | | Controller function create   // Find employee by ID  C:\xampp\htdocs\1\_api\API\_Provider\application\controllers\ApiDemo.php | // Find employee by ID      public function findEmp\_get($id)      {          $result\_find = $this->Employee\_model->findEmp($id);          if($result\_find) {              $this->response($result\_find, RestController::HTTP\_OK); // 200 OK          } else {              $this->response([                  'status' => false,                  'message' => 'Employee not found'              ], RestController::HTTP\_NOT\_FOUND); // 404 Not Found          }      } | |
|  | | Call through postman | <http://localhost:8070/api/find/2>  <http://localhost/1_api/API_Provider/index.php/api/find/10> | |
|  | |  |  | |
|  | |  |  | |
|  | | **To** Find employee by ID **through Second Project (API Consumer) into the First Project (API Provider) via API, you will need to create a controller that sends data using an HTTP POST request.** | | |
|  | |  | <http://localhost/1_api/API_Consumer/client/get_users> | |
|  | | Inserting Data **Second Project (API Consumer)** |  | |
|  | | Method to fetch employee details by ID  C:\xampp\htdocs\1\_api\API\_Consumer\application\controllers\ApiClientController.php | // Method to fetch employee details by ID      public function getEmployeeById($id) {          // API URL of the First Project (API Provider)          $api\_url = 'http://localhost/1\_api/API\_Provider/index.php/api/find/' . $id;            // Sending GET request to the API provider          $response = $this->curl->simple\_get($api\_url);          if ($response) {              // Decode the JSON response              $data['employee'] = json\_decode($response, true);              // Load the view to display the employee details              $this->load->view('employee\_details', $data);          } else {              // Handle failure case              $data['error'] = 'Failed to retrieve employee details.';              $this->load->view('error\_view', $data);          }      } | |
|  | | // Load the view to display the employee details  C:\xampp\htdocs\1\_api\API\_Consumer\application\views\employee\_details.php | <!-- application/views/employee\_view.php -->  <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Employee Details</title>      <style>          body {              font-family: Arial, sans-serif;              background-color: #f8d7da;              margin: 0;              padding: 0;              display: flex;              align-items: center;              justify-content: center;              height: 100vh;          }          .employee-container {              text-align: center;              background-color: #ffffff;              padding: 20px;              border-radius: 10px;              box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);          }          h1 {              font-size: 2em;              margin-bottom: 10px;          }          p {              font-size: 1.2em;              margin-bottom: 5px;          }      </style>  </head>  <body>      <div class="employee-container">          <h1>Employee Details</h1>          <p><strong>First Name:</strong> <?php echo $employee['first\_name']; ?></p>          <p><strong>Last Name:</strong> <?php echo $employee['last\_name']; ?></p>          <p><strong>Phone:</strong> <?php echo $employee['phone']; ?></p>          <p><strong>Email:</strong> <?php echo $employee['email']; ?></p>          <button><a href="<?php echo base\_url().'client/get\_users'; ?>">View All Emplyee</a></button>      </div>      </body>  </html> | |
|  | | // Handle failure case  C:\xampp\htdocs\1\_api\API\_Consumer\application\views\error\_view.php | <!-- application/views/error\_view.php -->  <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Error Page</title>      <style>          body {              font-family: Arial, sans-serif;              background-color: #f8d7da;              color: #721c24;              margin: 0;              padding: 0;              display: flex;              align-items: center;              justify-content: center;              height: 100vh;          }          .error-container {              text-align: center;              background-color: #f5c6cb;              padding: 20px;              border-radius: 10px;              box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);          }          h1 {              font-size: 2em;              margin-bottom: 10px;          }          p {              font-size: 1.2em;          }          a {              color: #721c24;              text-decoration: none;              font-weight: bold;          }          a:hover {              text-decoration: underline;          }      </style>  </head>  <body>      <div class="error-container">          <h1>Error!</h1>            <p><a href="<?php echo base\_url().'client/get\_users'; ?>">Go Back</a></p>      </div>  </body>  </html> | |
|  | | Recreate User View :  C:\xampp\htdocs\1\_api\API\_Consumer\application\views\users\_view.php | <!DOCTYPE html>  <html>  <head>      <title>Users List</title>      <style>          table {              width: 100%;              border-collapse: collapse;          }          table, th, td {              border: 1px solid black;          }          th, td {              padding: 8px;              text-align: left;          }          th {              background-color: #f2f2f2;          }      </style>  </head>  <body>      <h1>List of Users  <button><a href="<?php echo base\_url().'client/store'; ?>">Add Employee</a></button></h1>      <?php if (!empty($users)) : ?>          <table>              <thead>                  <tr>                      <th>ID</th>                      <th>First Name</th>                      <th>Last Name</th>                      <th>Phone</th>                      <th>Email</th>                      <th> Emplyee Details </th>                  </tr>              </thead>              <tbody>                  <?php foreach ($users as $user) : ?>                      <tr>                          <td><?php echo htmlspecialchars($user['id']); ?></td>                          <td><?php echo htmlspecialchars($user['first\_name']); ?></td>                          <td><?php echo htmlspecialchars($user['last\_name']); ?></td>                          <td><?php echo htmlspecialchars($user['phone']); ?></td>                          <td><?php echo htmlspecialchars($user['email']); ?></td>                            <td><button><a href="<?php echo base\_url().'get-employee/'.$user['id']; ?>">View This Emplyee</a></button></td>                        </tr>                  <?php endforeach; ?>              </tbody>          </table>      <?php else : ?>          <p>No users found.</p>      <?php endif; ?>  </body>  </html> | |
|  | | Router pert :  C:\xampp\htdocs\1\_api\API\_Consumer\application\config\routes.php | $route['client/get\_users'] = 'ApiClientController/get\_users';  $route['client/store'] = 'ApiClientController/form';  $route['get-employee/(:num)'] = 'ApiClientController/getEmployeeById/$1'; | |
|  | | C:\xampp\htdocs\1\_api\API\_Consumer\application\config\autoload.php | $autoload['helper'] = array('url');  $autoload['libraries'] = array('curl','session'); | |
|  | | C:\xampp\htdocs\1\_api\API\_Consumer\application\config\config.php | $config['base\_url'] = 'http://localhost/1\_api/API\_Consumer/'; | |
|  | | C:\xampp\htdocs\1\_api\API\_Consumer\.htaccess  Add the file | RewriteEngine on  RewriteCond %{REQUEST\_FILENAME} !-f  RewriteCond %{REQUEST\_FILENAME} !-d  RewriteRule .\* index.php/$0 [PT,L] | |
|  | |  | | |
|  | | Add Employee: |  | |
|  | |  | | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | | Specific data call Data (**Find employee by ID)** |  | |
|  | | 1. First Project (API Provider) | First Project (API Provider)  Specific data call Data (**Find employee by ID)** | |
|  | |  |  | |
|  | | Route create :  C:\xampp\htdocs\1\_api\API\_Provider\application\config\routes.php |  | |
|  | | Model function create specific dat calling  C:\xampp\htdocs\1\_api\API\_Provider\application\models\Employee\_model.php | public function findEmp($id)      {          $this->db->where('id',$id);          $query = $this->db->get('employee');          return $query->row();      } | |
|  | | Controller function create   // Find employee by ID  C:\xampp\htdocs\1\_api\API\_Provider\application\controllers\ApiDemo.php | // Find employee by ID      public function findEmp\_get($id)      {          $result\_find = $this->Employee\_model->findEmp($id);          if($result\_find) {              $this->response($result\_find, RestController::HTTP\_OK); // 200 OK          } else {              $this->response([                  'status' => false,                  'message' => 'Employee not found'              ], RestController::HTTP\_NOT\_FOUND); // 404 Not Found          }      } | |
|  | | Call through postman | <http://localhost:8070/api/find/2>  <http://localhost/1_api/API_Provider/index.php/api/find/10> | |
|  | |  |  | |
|  | | **To** Find employee by ID **through Second Project (API Consumer) into the First Project (API Provider) via API, you will need to create a controller that sends data using an HTTP POST request.** | | |
|  | |  | <http://localhost/1_api/API_Consumer/client/get_users> | |
| **Step:13** | | **Find employee by ID** |  | |
|  | | **Second Project (API Consumer)** | **Find employee by ID**  **Second Project (API Consumer) with Autication session** | |
|  | | Method to fetch employee details by ID  C:\xampp\htdocs\1\_api\API\_Consumer\application\controllers\ApiClientController.php  C:\xampp\htdocs\1\_api\API\_Consumer\_p2\application\controllers\ApiClientController.php | public function getEmployeeById($id)  {      // Check if API credentials are stored in session      if (!$this->session->userdata('api\_key') || !$this->session->userdata('username') || !$this->session->userdata('password')) {          $this->session->set\_flashdata('error', 'Please authenticate first');          redirect('ApiClientController/api\_key\_form');      }      // API URL of the First Project (API Provider)      $api\_url = 'http://localhost/1\_api/API\_Provider\_p2/index.php/api/find/' . $id;      // Retrieve authentication data from session      $api\_key = $this->session->userdata('api\_key');      $username = $this->session->userdata('username');      $password = $this->session->userdata('password');      // Initialize cURL      $ch = curl\_init($api\_url);      // Set cURL options      curl\_setopt($ch, CURLOPT\_RETURNTRANSFER, true);      curl\_setopt($ch, CURLOPT\_HTTPHEADER, array(          'Content-Type: application/json',          'X-API-KEY: ' . $api\_key // Include API key in the header      ));      // Set Basic Authentication credentials      curl\_setopt($ch, CURLOPT\_USERPWD, $username . ":" . $password); // Basic Auth      curl\_setopt($ch, CURLOPT\_HTTPAUTH, CURLAUTH\_BASIC);      // Execute the cURL request      $response = curl\_exec($ch);      // Check for cURL errors      if (curl\_errno($ch)) {          $error = curl\_error($ch);          curl\_close($ch);          // Handle failure case and load error view          $data['error'] = 'Request Error: ' . $error;          $this->load->view('error\_view', $data);      } else {          curl\_close($ch);          // Decode the JSON response          $data['employee'] = json\_decode($response, true);          // Check if the employee data is found          if (!empty($data['employee'])) {              // Load the view to display the employee details              $this->load->view('employee\_view', $data);          } else {              // Handle employee not found              $data['error'] = 'Employee not found.';              $this->load->view('error\_view', $data);          }      }  } | |
|  | | // Load the view to display the employee details  C:\xampp\htdocs\1\_api\API\_Consumer\application\views\employee\_details.php  C:\xampp\htdocs\1\_api\API\_Consumer\_p2\application\views\employee\_view.php | <!-- application/views/employee\_view.php -->  <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Employee Details</title>      <style>          body {              font-family: Arial, sans-serif;              background-color: #f8d7da;              margin: 0;              padding: 0;              display: flex;              align-items: center;              justify-content: center;              height: 100vh;          }          .employee-container {              text-align: center;              background-color: #ffffff;              padding: 20px;              border-radius: 10px;              box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);          }          h1 {              font-size: 2em;              margin-bottom: 10px;          }          p {              font-size: 1.2em;              margin-bottom: 5px;          }      </style>  </head>  <body>      <div class="employee-container">          <h1>Employee Details</h1>          <p><strong>Employee ID:</strong> <?php echo $employee['id']; ?></p>          <p><strong>First Name:</strong> <?php echo $employee['first\_name']; ?></p>          <p><strong>Last Name:</strong> <?php echo $employee['last\_name']; ?></p>          <p><strong>Phone:</strong> <?php echo $employee['phone']; ?></p>          <p><strong>Email:</strong> <?php echo $employee['email']; ?></p>          <button><a href="<?php echo base\_url().'client/get\_users'; ?>">View All Emplyee</a></button>      </div>      </body>  </html> | |
|  | |  |  | |
|  | | Recreate User View :  C:\xampp\htdocs\1\_api\API\_Consumer\application\views\users\_view.php                      <th> Emplyee Details </th>  <td><button><a href="<?php echo base\_url().'get-employee/'.$user['id']; ?>">View This Emplyee</a></button></td> | <!DOCTYPE html>  <html>  <head>      <title>Users List</title>      <style>          table {              width: 100%;              border-collapse: collapse;          }          table, th, td {              border: 1px solid black;          }          th, td {              padding: 8px;              text-align: left;          }          th {              background-color: #f2f2f2;          }      </style>  </head>  <body>      <h1>List of Users  <button><a href="<?php echo base\_url().'client/store'; ?>">Add Employee</a></button></h1>      <?php if (!empty($users)) : ?>          <table>              <thead>                  <tr>                      <th>ID</th>                      <th>First Name</th>                      <th>Last Name</th>                      <th>Phone</th>                      <th>Email</th>                      <th> Emplyee Details </th>                  </tr>              </thead>              <tbody>                  <?php foreach ($users as $user) : ?>                      <tr>                          <td><?php echo htmlspecialchars($user['id']); ?></td>                          <td><?php echo htmlspecialchars($user['first\_name']); ?></td>                          <td><?php echo htmlspecialchars($user['last\_name']); ?></td>                          <td><?php echo htmlspecialchars($user['phone']); ?></td>                          <td><?php echo htmlspecialchars($user['email']); ?></td>                            <td><button><a href="<?php echo base\_url().'get-employee/'.$user['id']; ?>">View This Emplyee</a></button></td>                        </tr>                  <?php endforeach; ?>              </tbody>          </table>      <?php else : ?>          <p>No users found.</p>      <?php endif; ?>  </body>  </html> | |
|  | | Router pert :  C:\xampp\htdocs\1\_api\API\_Consumer\application\config\routes.php | $route['get-employee/(:num)'] = 'ApiClientController/getEmployeeById/$1'; | |
|  | |  | <http://localhost/1_api/API_Consumer/client/get_users> | |
|  | |  | <http://localhost/1_api/API_Consumer_p2/client/get_users>  http://localhost/1\_api/API\_Consumer\_p2/get-employee/3 | |
|  | |  |  | |
|  | |  | | |
|  | |  |  | |
|  | |  |  | |
|  | | $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$ | $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$ | |
| STEP :14 | | update employee by ID: | **To** update employee by ID **through Second Project (API Consumer) into the First Project (API Provider) via API . please provide code** | |
|  | |  | 1. First Project (API Provider) | |
|  | |  | C:\xampp\htdocs\1\_api\API\_Provider\application\controllers\ApiDemo.php  C:\xampp\htdocs\1\_api\API\_Provider\_p2\application\controllers\ApiDemo.php | |
|  | | 1. First Project (API Provider) | Update data () | |
|  | |  |  | |
|  | | Update raw data :  C:\xampp\htdocs\1\_api\API\_Provider\application\controllers\ApiDemo.php | public function updateEmp\_put($id)      {          $emp = new Employee\_model;          $data=[              'first\_name'=> $this->put('first\_name'),              'last\_name'=> $this->put('last\_name'),              'phone'=> $this->put('phone'),              'email'=> $this->put('email'),          ];          $update\_result =$emp ->update\_employee($id,$data);          // $this ->response($data,$resul,200);          if($update\_result>0)          {              $this->response([                  'status'=> true,                  'message'=> 'New Employee updated'              ] ,RestController::HTTP\_OK );          }          else          {              $this->response([                  'status'=> false,                  'message'=> 'Failed to  update employee'              ] ,RestController::HTTP\_BAD\_REQUEST );          }      } | |
|  | | route create  C:\xampp\htdocs\1\_api\API\_Provider\application\config\routes.php | $route['api/update/(:any)'] = 'ApiDemo/updateEmp/$1'; | |
|  | | model function create  C:\xampp\htdocs\1\_api\API\_Provider\application\models\Employee\_model.php | public function update\_employee($id,$data)      {          $this->db->where('id',$id);          return  $this->db->update('employee',$data);      } | |
|  | |  |  | |
|  | | Call postman  <http://localhost/1_api/API_Provider/index.php/api/update/1>  text =JSON \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* | method :put  <http://localhost/1_api/API_Provider/index.php/api/update/1>  <http://localhost/1_api/API_Provider_p2/index.php/api/update/7>  body /raw  {  "first\_name": "Art Of Living",  "last\_name": "Internal Center",  "phone": "5656565656",  "email": "jayravi@artofliving.org"  }  text =JSON \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* | |
|  | |  | **To** update employee by ID **through Second Project (API Consumer) into the First Project (API Provider) via API . please provide code** | |
|  | |  |  | |
|  | |  |  | |
|  | | C:\xampp\htdocs\1\_api\API\_Provider\.htaccess  Add the file.  .htaccess | RewriteEngine on  RewriteCond %{REQUEST\_FILENAME} !-f  RewriteCond %{REQUEST\_FILENAME} !-d  RewriteRule .\* index.php/$0 [PT,L] | |
|  | |  | <http://localhost/1_api/API_Provider/api/update/12> | |
|  | | Call the page again without index.php  <http://localhost/1_api/API_Provider/api/update/12> |  | |
|  | | **To** update employee by ID **through Second Project (API Consumer) into the First Project (API Provider) via API, you will need to create a controller that sends data using an HTTP POST request.** | | | | |
|  | |  | | |  | |
|  | | **Second Project (API Consumer)** | | | update employee by ID | |
|  | | C:\xampp\htdocs\1\_api\API\_Consumer\application\config\routes.php  Rote create | | | $route['load-update-form'] = 'ApiClientController/load\_update\_form';  // Route to load the update form  $route['update-employee'] = 'ApiClientController/update\_employee';    // Route to handle form submission | |
|  | |  | | | public function load\_update\_form()      {          $this->load->view('update\_employee\_view'); // Load the form view      }      // Function to handle form submission and update employee via API      public function update\_employee()      {          // Get form data          $id = $this->input->post('id');          $first\_name = $this->input->post('first\_name');          $last\_name = $this->input->post('last\_name');          $phone = $this->input->post('phone');          $email = $this->input->post('email');          // Prepare data to send to the API Provider          $data = array(              'first\_name' => $first\_name,              'last\_name' => $last\_name,              'phone' => $phone,              'email' => $email          );          // API endpoint (URL of the First Project's API)          $url = "http://localhost/1\_api/API\_Provider/index.php/api/update/" . $id;          // Initialize cURL          $ch = curl\_init($url);          // Convert data array to JSON          $json\_data = json\_encode($data);          // Set cURL options for a PUT request          curl\_setopt($ch, CURLOPT\_RETURNTRANSFER, true);          curl\_setopt($ch, CURLOPT\_CUSTOMREQUEST, "PUT");          curl\_setopt($ch, CURLOPT\_POSTFIELDS, $json\_data);          curl\_setopt($ch, CURLOPT\_HTTPHEADER, array(              'Content-Type: application/json',              'Content-Length: ' . strlen($json\_data)          ));          // Execute the cURL request          $response = curl\_exec($ch);          // Check for cURL errors          if (curl\_errno($ch)) {              $error = curl\_error($ch);              curl\_close($ch);              // Load error view if there is an issue              $data['message'] = $error;              $this->load->view('error\_view', $data);          } else {              curl\_close($ch);              // Decode the API response              $api\_response = json\_decode($response, true);              // Check the response status from the API              if ($api\_response['status'] === true) {                  $this->session->set\_flashdata('success', $api\_response['message']);                  redirect('client/get\_users'); // Redirect to the URL you specified              } else {                  $this->session->set\_flashdata('error', $api\_response['message']);                  redirect('client/error'); // Redirect to an error page              }          }      } | |
|  | | View Page Create  C:\xampp\htdocs\1\_api\API\_Consumer\application\views\update\_employee\_view.php | | | <!-- application/views/update\_employee\_view.php -->  <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Update Employee</title>      <style>          body {              font-family: Arial, sans-serif;              background-color: #f4f7f8;              margin: 0;              padding: 0;              display: flex;              justify-content: center;              align-items: center;              height: 100vh;          }          .container {              background-color: #fff;              padding: 20px;              border-radius: 10px;              box-shadow: 0px 0px 15px rgba(0, 0, 0, 0.1);          }          h1 {              color: #333;              text-align: center;          }          form {              display: flex;              flex-direction: column;              gap: 10px;          }          label {              font-weight: bold;          }          input {              padding: 10px;              border: 1px solid #ccc;              border-radius: 5px;          }          button {              padding: 10px;              background-color: #4CAF50;              color: white;              border: none;              border-radius: 5px;              cursor: pointer;          }          button:hover {              background-color: #45a049;          }      </style>  </head>  <body>      <div class="container">          <h1>Update Employee</h1>          <form action="<?php echo base\_url('ApiClientController/update\_employee'); ?>" method="POST">              <label for="id">Employee ID:</label>              <input type="text" id="id" name="id" required>              <label for="first\_name">First Name:</label>              <input type="text" id="first\_name" name="first\_name" required>              <label for="last\_name">Last Name:</label>              <input type="text" id="last\_name" name="last\_name" required>              <label for="phone">Phone:</label>              <input type="text" id="phone" name="phone" required>              <label for="email">Email:</label>              <input type="email" id="email" name="email" required>              <button type="submit">Update Employee</button>          </form>      </div>  </body>  </html> | |
|  | | User view updated :  C:\xampp\htdocs\1\_api\API\_Consumer\application\views\users\_view.php | | | <!DOCTYPE html>  <html>  <head>      <title>Users List</title>      <style>          table {              width: 100%;              border-collapse: collapse;          }          table, th, td {              border: 1px solid black;          }          th, td {              padding: 8px;              text-align: left;          }          th {              background-color: #f2f2f2;          }      </style>  </head>  <body>      <h1>List of Users  <button><a href="<?php echo base\_url().'client/store'; ?>">Add Employee</a></button></h1>      <?php if (!empty($users)) : ?>          <table>              <thead>                  <tr>                      <th>Emplyee ID</th>                      <th>First Name</th>                      <th>Last Name</th>                      <th>Phone</th>                      <th>Email</th>                      <th> Emplyee Details </th>                      <th> Update Emplyee  </th>                  </tr>              </thead>              <tbody>                  <?php foreach ($users as $user) : ?>                      <tr>                          <td><?php echo htmlspecialchars($user['id']); ?></td>                          <td><?php echo htmlspecialchars($user['first\_name']); ?></td>                          <td><?php echo htmlspecialchars($user['last\_name']); ?></td>                          <td><?php echo htmlspecialchars($user['phone']); ?></td>                          <td><?php echo htmlspecialchars($user['email']); ?></td>                            <td><button><a href="<?php echo base\_url().'get-employee/'.$user['id']; ?>">View This Emplyee</a></button></td>                          <td><button><a href="<?php echo base\_url().'load-update-form/'; ?>">Update Emplyee</a></button></td>                        </tr>                  <?php endforeach; ?>              </tbody>          </table>      <?php else : ?>          <p>No users found.</p>      <?php endif; ?>  </body>  </html> | |
|  | |  | | | method :put  <http://localhost/1_api/API_Consumer/load-update-form/> | |
|  | | To implement a delete operation for employee data through your API and then handle it via the view in your Second Project (API Consumer), follow these steps: | | | | |
|  | | Delete : | | |  | |
|  | |  | | | * **Complete the merge:**   bash  Copy code  git commit -m "Resolved merge conflicts and completed the merge"  git push origin master   * **(Optional) Rebase to address divergence:**   bash  Copy code  git pull --rebase origin master  git push origin master  By following these steps, you will complete the merge and address any divergence between your local and remote branches. | |
|  | |  | | | 1. First Project (API Provider) | |
|  | | 1. First Project (API Provider) | | | Delete Data: | |
|  | | Route create For delete:  C:\xampp\htdocs\1\_api\API\_Provider\application\config\routes.php | | | $route['api/delete/(:any)'] = 'ApiDemo/deleteEmp/$1'; | |
|  | | Delete method in Controller :  C:\xampp\htdocs\1\_api\API\_Provider\application\controllers\ApiDemo.php | | | public function deleteEmp\_delete($id)      {          $emp = new Employee\_model;          $result = $emp ->delete\_emp($id);          if($result>0)          {              $this->response([                  'status'=> true,                  'message'=> 'New Employee deleted'              ] ,RestController::HTTP\_OK );          }          else          {              $this->response([                  'status'=> false,                  'message'=> 'Failed to  Delete employee'              ] ,RestController::HTTP\_BAD\_REQUEST );          }      } | |
|  | | Delete function create in model  C:\xampp\htdocs\1\_api\API\_Provider\application\models\Employee\_model.php | | | public function delete\_emp($id)      {          $this->db->where('id',$id);          return  $this->db->delete('employee',['id'=>$id]);      } | |
|  | |  | | | Call through postaman | |
|  | | Delete the data | | | <http://localhost/1_api/API_Provider/api/delete/12> | |
|  | |  | | |  | |
|  | | Call through postman | | |  | |
|  | |  |  | |
|  | |  |  | |
|  | | **Second Project (API Consumer)** | Update the API Consumer: | |
|  | |  |  | |
|  | | Route create For delete:  C:\xampp\htdocs\1\_api\API\_Consumer\application\config\routes.php | $route['delete\_employee/(:num)'] = 'ApiClientController/delete\_employee/$1';// Route to handle form submission | |
|  | | <td><button><a href="<?php echo base\_url().'delete\_employee/'.$user['id']; ?>">Delete This Emplyee</a></button></td>                      <th> Delete Emplyee  </th>  Add this tow line  C:\xampp\htdocs\1\_api\API\_Consumer\application\views\users\_view.php | <table>              <thead>                  <tr>                      <th>Emplyee ID</th>                      <th>First Name</th>                      <th>Last Name</th>                      <th>Phone</th>                      <th>Email</th>                      <th> Emplyee Details </th>                      <th> Update Emplyee  </th>                      <th> Delete Emplyee  </th>                  </tr>              </thead>              <tbody>                  <?php foreach ($users as $user) : ?>                      <tr>                          <td><?php echo htmlspecialchars($user['id']); ?></td>                          <td><?php echo htmlspecialchars($user['first\_name']); ?></td>                          <td><?php echo htmlspecialchars($user['last\_name']); ?></td>                          <td><?php echo htmlspecialchars($user['phone']); ?></td>                          <td><?php echo htmlspecialchars($user['email']); ?></td>                            <td><button><a href="<?php echo base\_url().'get-employee/'.$user['id']; ?>">View This Emplyee</a></button></td>                          <td><button><a href="<?php echo base\_url().'load-update-form/'; ?>">Update Emplyee</a></button></td>                          <td><button><a href="<?php echo base\_url().'delete\_employee/'.$user['id']; ?>">Delete This Emplyee</a></button></td>                        </tr>                  <?php endforeach; ?>              </tbody>          </table> | |
|  | | HTML with Styled Buttons::  C:\xampp\htdocs\1\_api\API\_Consumer\_p2\application\views\users\_view.php | <td>                              <a href="<?php echo base\_url() . 'get-employee/' . $user['id']; ?>" class="btn view-btn">View This Employee</a>                          </td>                          <td>                              <a href="<?php echo base\_url() . 'load-update-form/' . $user['id']; ?>" class="btn update-btn">Update Employee</a>                          </td>                          <td>                              <a href="<?php echo base\_url() . 'delete\_employee/' . $user['id']; ?>" class="btn delete-btn">Delete This Employee</a>                          </td> | |
|  | |  | <style>          .btn {              display: inline-block;              padding: 10px 20px;              margin: 5px;              text-align: center;              text-decoration: none;              font-size: 16px;              border-radius: 5px;              transition: background-color 0.3s ease;          }          .view-btn {              background-color: #4CAF50;              color: white;          }          .update-btn {              background-color: #FF9800;              color: white;          }          .delete-btn {              background-color: #f44336;              color: white;          }          /\* Hover effects \*/          .btn:hover {              opacity: 0.8;          }          /\* Optional: making the links look like buttons \*/          .btn a {              color: inherit;              text-decoration: none;          }      </style> | |
|  | | Delete function create :  C:\xampp\htdocs\1\_api\API\_Consumer\application\controllers\ApiClientController.php | public function delete\_employee($id)      {          // API endpoint (URL of the First Project's API)          $url = "http://localhost/1\_api/API\_Provider/index.php/api/delete/" . $id;          // Initialize cURL          $ch = curl\_init($url);          // Set cURL options for a DELETE request          curl\_setopt($ch, CURLOPT\_RETURNTRANSFER, true);          curl\_setopt($ch, CURLOPT\_CUSTOMREQUEST, "DELETE");          // Execute the cURL request          $response = curl\_exec($ch);          // Check for cURL errors          if (curl\_errno($ch)) {              $error = curl\_error($ch);              curl\_close($ch);              // Redirect to an error page with an error message              $this->session->set\_flashdata('error', $error);              redirect('client/error');          } else {              curl\_close($ch);              // Decode the API response              $api\_response = json\_decode($response, true);              // Check the response status from the API              if ($api\_response['status'] === true) {                  $this->session->set\_flashdata('success', $api\_response['message']);                  redirect('client/get\_users'); // Redirect to the URL you specified              } else {                  $this->session->set\_flashdata('error', $api\_response['message']);                  redirect('client/error'); // Redirect to an error page              }          }      } | |
|  | |  |  | |
|  | | API \_provider : | Server setup | |
|  | |  |  | |
|  | | #[\AllowDynamicProperties] | Message: Creation of dynamic property CI\_URI::$config is deprecated  C:\xampp\htdocs\1\_api\API\_Consumer\_p2\system\core  Filename: core/URI.php  Filename: core/Router.php  Filename: core/Controller.php  Filename: core/Loader.php  Function: require\_once | |
|  | |  |  | |
|  | | To configure your CodeIgniter project for both localhost and the live website C:\xampp\htdocs\1\_api\pgination\_CI\application\config\config.php | // $config['base\_url'] = '';  if ($\_SERVER['HTTP\_HOST'] == 'localhost') {      $config['base\_url'] = 'http://localhost/1\_api/API\_Provider/';  } else {      $config['base\_url'] = 'https://krishnendudalui.in.net/API\_Provider/';  } | |
|  | |  |  | |
|  | | **To manage database connections for both local and live environments:** | **Code igniter frame work:** | |
|  | |  | $active\_group = 'default';  $query\_builder = TRUE;  // Check environment dynamically (local/live)  $environment = ($\_SERVER['SERVER\_NAME'] === 'localhost') ? 'local' : 'live';  if ($environment === 'local') {      // Local settings      $db['default'] = array(          'dsn'      => '',          'hostname' => 'localhost',          'username' => 'root',          'password' => '',          'database' => 'busbooking',          'dbdriver' => 'mysqli',          'dbprefix' => '',          'pconnect' => FALSE,          'db\_debug' => (ENVIRONMENT !== 'production'),          'cache\_on' => FALSE,          'cachedir' => '',          'char\_set' => 'utf8',          'dbcollat' => 'utf8\_general\_ci',          'swap\_pre' => '',          'encrypt' => FALSE,          'compress' => FALSE,          'stricton' => FALSE,          'failover' => array(),          'save\_queries' => TRUE      );  } else {      // Live server settings      $db['default'] = array(          'dsn'      => '',          'hostname' => 'localhost', // Can be your live database endpoint (EC2, RDS, etc.)          'username' => 'root', // Live MySQL username          'password' => 'D@2015$yuti', // Live MySQL password          'database' => 'busbooking',          'dbdriver' => 'mysqli',          'dbprefix' => '',          'pconnect' => FALSE,          'db\_debug' => (ENVIRONMENT !== 'production'),          'cache\_on' => FALSE,          'cachedir' => '',          'char\_set' => 'utf8',          'dbcollat' => 'utf8\_general\_ci',          'swap\_pre' => '',          'encrypt' => FALSE,          'compress' => FALSE,          'stricton' => FALSE,          'failover' => array(),          'save\_queries' => TRUE      );  } | |
|  | | Data Base update | <https://krishnendudalui.in.net/phpmyadmin/> | |
|  | |  |  | |
|  | |  |  | |
|  | | Correct Steps to Clone a Git Repository cd /var/www/html/API\_Provider  git pull  cd API\_Provider | Correct Steps to Clone a Git Repository  1. **Navigate to the Target Directory:**   bash  Copy code  cd /var/www/html/   1. **Use the git clone Command:**   bash  Copy code  git clone https://github.com/Code-with-nandu/API\_Provider.git  This will clone the repository into the /var/www/html/API\_Provider directory.   1. Update   Git pull | |
|  | | Call the page | <https://krishnendudalui.in.net/API_Provider/> | |
|  | | Call through postman | <https://krishnendudalui.in.net/API_Provider/api> | |
|  | | Error came | The error you're encountering in your PHP application is due to a call to the strtolower() function, which is receiving a null value instead of a string. This is a deprecation warning that indicates future versions of PHP will not allow passing null to this function.  **How to Fix the Error**   1. **Identify the Source of null:** In your case, the error is happening in **RestController.php** on line 835. You need to look at the code around that line to understand where the variable being passed to strtolower() is coming from. 2. **Add a Check for Null:** Before calling strtolower(), you can check if the variable is null. If it is, you can either set a default value or handle it in some wa | |
|  | | Resolution.  C:\xampp\htdocs\1\_api\API\_Provider\application  \libraries\RestController.php | protected function \_detect\_method()  {      // Declare a variable to store the method      $method = null;      // Determine whether the 'enable\_emulate\_request' setting is enabled      if ($this->config->item('enable\_emulate\_request') === true) {          $method = $this->input->post('\_method');            if ($method === null) {              $method = $this->input->server('HTTP\_X\_HTTP\_METHOD\_OVERRIDE');          }          // Check if method is not null before calling strtolower          if ($method !== null) {              $method = strtolower($method);          }      }      if (empty($method)) {          // Get the request method as a lowercase string          $method = $this->input->method();      }      // Ensure the method is a valid string before proceeding      if (!is\_string($method)) {          $method = 'get'; // Default to 'get' if it's not valid      }      return in\_array($method, $this->allowed\_http\_methods) && method\_exists($this, '\_parse\_'.$method) ? $method : 'get';  } | |
|  | | Call the page for whole data: | <https://krishnendudalui.in.net/API_Provider/api>  <http://localhost/1_api/API_Provider/index.php/api> | |
|  | |  |  | |
|  | |  |  | |
|  | | Specific call through id | <https://krishnendudalui.in.net/API_Provider/api/find/9>  <http://localhost/1_api/API_Provider/index.php/api/find/9> | |
|  | |  |  | |
|  | | Update data thorough put method | <https://krishnendudalui.in.net/API_Provider/api/update/9>  <http://localhost/1_api/API_Provider/index.php/api/update/9> | |
|  | | {          "first\_name": "Art Of Living1",          "last\_name": "Internal Center1",          "phone": "56565656561",          "email": "jayravi@artofliving.org1"      } |  | |
|  | | Data added in data based : | <https://krishnendudalui.in.net/API_Provider/api/store>  <http://localhost/1_api/API_Provider/api/store> | |
|  | |  |  | |
|  | | Delete method : | <http://localhost/1_api/API_Provider/api/delete/12>  <https://krishnendudalui.in.net/API_Provider/api/delete/12> | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | | |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | Sl | Task | Sub task | alocated | status |  |  | | 2 | live and local check api provider V1 | 1. task create | 15 minit | Done | 7.15 | 7.3 | |  |  | STEP:1 | 30 minits | running | 7.3 | 8 | |  |  | 2. GET, PUT , POST , DELTE | 5 minit | done |  |  | |  |  | a. call all data | 5 minit | done |  |  | |  |  | b. call spefic data | 5 minit | done |  |  | |  |  | c. insert data | 5 minit | done |  |  | |  |  | d. modify data | 5 minit | done |  |  | |  |  | e. delete data | 5 minit | done |  |  | |  |  | STEP 2 | 20 minits |  | 09.00 pm | 09.20 pm | |  |  | a. Upload goole drive for postman | 5 minit | done |  |  | |  |  | b. Upload goole drive\_whole doc upload | 5 minit | done |  |  | |  |  | c. create link for welcome .php | 5 minit | done |  |  | |  |  | d. create link for b welcome .php | 5 minit | done |  |  | |  |  | STEP 3 | 15 minits |  | 09.20 pm | 9.35 pm | |  |  | a . Update in live | 5 minit | done |  |  | |  |  | b. call throgh local | 5 minit | done |  |  | |  |  | c upadte in gurudev | 5 minit | Done |  |  | |  |  |  |  |  |  |  | |  |  | STEP 4 | 10 mints |  | 9.35 pm | 09.55 pm | |  |  | review the task | 10 minit |  |  |  | | | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | | API \_consumer : |  | |
|  | | Server setup |  | |
|  | |  |  | |
|  | | #[\AllowDynamicProperties] | Message: Creation of dynamic property CI\_URI::$config is deprecated  C:\xampp\htdocs\1\_api\API\_Consumer\_p2\system\core  Filename: core/URI.php  Filename: core/Router.php  Filename: core/Controller.php  Filename: core/Loader.php  Function: require\_once | |
|  | | API\_Consumer |  | |
|  | | To configure your CodeIgniter project for both localhost and the live website C:\xampp\htdocs\1\_api\API\_Consumer\application\config\config.php | // $config['base\_url'] = 'http://localhost/1\_api/API\_Consumer/';  // $config['base\_url'] = '';  if ($\_SERVER['HTTP\_HOST'] == 'localhost') {      $config['base\_url'] = 'http://localhost/1\_api/API\_Consumer/';  } else {      $config['base\_url'] = 'https://krishnendudalui.in.net/API\_Consumer/';  }  /\*  Change to 'production' for live environment  \*/  $config['environment'] = 'development';  /\* | |
|  | | Correct Steps to Clone a Git Repository cd /var/www/html/API\_Consumer  git pull  cd API\_Provider | Correct Steps to Clone a Git Repository  1. **Navigate to the Target Directory:**   bash  Copy code  cd /var/www/html/   1. **Use the git clone Command:**   bash  Copy code  git clone https://github.com/Code-with-nandu/API\_Consumer.git  This will clone the repository into the /var/www/html/API\_Provider directory.   1. Update   Git pull | |
|  | | Call the page |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | | To configure your CodeIgniter project for both localhost and the live website C:\xampp\htdocs\1\_api\API\_Consumer\application\config\config.php | // $config['base\_url'] = 'http://localhost/1\_api/API\_Consumer/';  // $config['base\_url'] = '';  if ($\_SERVER['HTTP\_HOST'] == 'localhost') {      $config['base\_url'] = 'http://localhost/1\_api/API\_Consumer/';  } else {      $config['base\_url'] = 'https://krishnendudalui.in.net/API\_Consumer/';  }  /\*  Change to 'production' for live environment  \*/  $config['environment'] = 'development';  /\* | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | | // API endpoint (URL of the First Project's API) | // API endpoint (URL of the First Project's API)          $url = $this->config->item('environment') === 'production'              ? "https://krishnendudalui.in.net/API\_Provider/index.php/api/delete/$id"              : "http://localhost/1\_api/API\_Provider/index.php/api/delete/$id"; | |
|  | |  |  | |
|  | |  | // Determine the base URL based on the environment          $this->base\_url = $this->config->item('environment') === 'production'              ? "https://krishnendudalui.in.net/API\_Provider"              : "http://localhost/1\_api/API\_Provider"; | |
|  | |  | // Initialize cURL          $ch = curl\_init("{$this->base\_url}/index.php/auth/login");          $json\_data = json\_encode($data); | |
|  | | C:\xampp\htdocs\1\_api\API\_Consumer\application\config\config.php | Change to 'production' for live environment  \*/  $config['environment'] = 'development';  /\* | |
|  | | Call the page  <https://krishnendudalui.in.net/API_Provider_p2/> | <http://localhost/1_api/API_Provider_p2/> | |
|  | |  |  | |
|  | | Remove all |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |
|  | |  |  | |