⑂ **7ef8bd8d60** ⌄   ···

rconfig / www / install / lib / ajaxHandlers / **ajaxDbInstall.php** / <> Jump to ⌄

🔶 **rconfig** remove echo from DBinstall script   ⟲ History

👥 **2 contributors** 🔶 🔴

116 lines (111 sloc) | 4.74 KB   ···

```php
  1  <?php
  2  error_reporting(E_ALL);
  3  ini_set('display_errors', 1);
  4
  5  if (isset($_GET['server'])) {
  6      $server = $_GET['server'];
  7  }
  8  if (isset($_GET['port'])) {
  9      $port = $_GET['port'];
 10  }
 11  if (isset($_GET['dbName'])) {
 12      $dbName = $_GET['dbName'];
 13  }
 14  if (isset($_GET['dbUsername'])) {
 15      $dbUsername = $_GET['dbUsername'];
 16  }
 17  if (isset($_GET['dbPassword'])) {
 18      $dbPassword = $_GET['dbPassword'];
 19  }
 20  $sqlHost = $server . ":" . $port;
 21  $dbFile = '../../rconfig.sql';
 22  $configFilePathOriginal = '/home/rconfig/www/install/config.inc.php.template';
 23  $configFilePathInstalled = '/home/rconfig/config/config.inc.php';
 24  $array = array();
 25
 26  // add DB
 27  $options = array(
 28      PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION,
 29      PDO::ATTR_PERSISTENT => true,
 30      PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC
 31  );
 32  $error = ''; // set error var to blank
 33  try {
 34      $dbh = new PDO('mysql:port=' . $port . ';host=' . $server, $dbUsername, $dbPassword, $options);
 35      $dbh->exec("CREATE DATABASE `$dbName`;");
 36      $dbCreated = 1;
 37  } catch (PDOException $e) {
 38      // delete DB if there was an error running the CREATE DB query
 39      try {
 40          $dbh = new PDO('mysql:port=' . $port . ';host=' . $server, $dbUsername, $dbPassword, $options);
 41          $dbh->exec("DROP DATABASE `$dbName`;");
 42      } catch (PDOException $e) {
 43          $error = "DB ERROR: " . $e->getMessage() . "(some special characters such as + signs are not allowed in MYSQl passwords)";
 44  //          echo $error;
 45          $array['error'] = '<strong><font class="bad">Fail - ' . $error . '</strong></font><br/>';
 46          echo json_encode($array);
 47          die();
 48      }
 49      $error = "DB ERROR: " . $e->getMessage();
 50      $array['error'] = '<strong><font class="bad">Fail - ' . $error . '</strong></font><br/>';
 51      echo json_encode($array);
 52      die();
 53  }
 54
 55  if ($dbCreated == 1) {
 56      // rewrite the 'DATABASE_NAME' tag from the SQL file into memory
 57      $templateFile = file_get_contents($dbFile);
 58      $templateFile = str_replace('DATABASE_NAME', $dbName, $templateFile);
 59      $sql = file_get_contents($dbFile); //file name should be name of SQL file with .sql extension.
 60      try {
 61          $dsn = 'mysql:host=' . $server . ';dbname=' . $dbName . ';port=' . $port;
 62          $dbh = new PDO($dsn, $dbUsername, $dbPassword, $options);
 63          $sqlArray = explode(';', $templateFile);
 64          foreach ($sqlArray as $stmt) { //loop through each line of the sql file and execute
 65              if (strlen($stmt) > 3 && substr(ltrim($stmt), 0, 2) != '/*') {
 66                  $sth = $dbh->prepare($stmt);
 67                  $sth->execute();
 68              }
 69          }
 70      } catch (PDOException $e) {
 71          // delete DB if there was an error running any commands
 72          try {
 73              $dbh = new PDO('mysql:port=' . $port . ';host=' . $server, $dbUsername, $dbPassword, $options);
 74              $dbh->exec("DROP DATABASE `$dbName`;");
 75          } catch (PDOException $e) {
 76              $error = "DB ERROR: " . $e->getMessage();
 77              echo $error;
 78              $array['error'] = '<strong><font class="bad">Fail - ' . $error . '</strong></font><br/>';
```

```php
            echo json_encode($array);
            die();
        }
        $error = "DB ERROR: " . $e->getMessage();
        $array['error'] = '<strong><font class="bad">Fail - ' . $error . '. We have deleted the DB that was created. Please check your settings and try again.</strong></font><br/>';
        echo json_encode($array);
        die();
    }
    /* Add details to /includes/config.inc.php file */
    $configFile = file_get_contents($configFilePathOriginal);
    // re-write config file in memory
    $configFile = str_replace('_DATABASEHOST', $server, $configFile);
    $configFile = str_replace('_DATABASEPORT', $port, $configFile);
    $configFile = str_replace('_DATABASENAME', $dbName, $configFile);
    $configFile = str_replace('_DATABASEUSERNAME', $dbUsername, $configFile);
    $configFile = str_replace('_DATABASEPASSWORD', $dbPassword, $configFile);

    chmod($configFilePathInstalled, 0777);
    file_put_contents($configFilePathInstalled, $configFile);
    chmod($configFilePathInstalled, 0644);
    shell_exec('chown -R apache /home/rconfig'); // set all dir permissions correctly

    $array['success'] = '<strong><font class="Good">rConfig database installed successfully</strong></font><br/>';
} else {
    $array['error'] = '<strong><font class="bad">Fail - ' . $error . '</strong></font><br/>';
    // DROP DB on failure
    try {
        $dbh = new PDO('mysql:port=' . $port . ';host=' . $server, $dbUsername, $dbPassword, $options);
        $dbh->exec("DROP DATABASE `$dbName`;");
    } catch (PDOException $e) {
        $error = "DB ERROR: " . $e->getMessage();
        echo $error;
        $array['error'] = '<strong><font class="bad">Fail - ' . $error . '</strong></font><br/>';
        echo json_encode($array);
        die();
    }
}
echo json_encode($array);
```