

Acceptance Testing Document

Input Sequence	Expected Result	Current Output	Comments
Req 1: Login			
Correct login: valid_name valid_password	Login accepted	Login accepted	Passed
Incorrect login: valid_name wrong_password	Login incorrect	Page refresh	Passed
Req 2: Search function			
Searches	Brings result	Brings result	Passed
Req 3: Update			
Updates student record	Updates student record	Updates student record	Passed
Updates supervisor record	Updates supervisor record	Updates supervisor record	Passed
Updates project record	Updates project record	Updates project record	Passed
Req 4: Delete			
Delete Student record	Delete Student record and user account	Delete Student record and user account	Passed
Delete Supervisor record	Delete supervisor record and user account	Delete supervisor record and user account	Passed
Delete project record	Delete project record	Delete project record	Passed
Req 5: Add			
Add Student record	Adds student record and user account	Adds student record and user account	Passed
Add Supervisor record	Adds supervisor record and user account	Adds supervisor record and user account	Passed
Add project record	Adds project record	Adds project record	Passed
Req 6: Sort Ascending/Descending			
Sorts table ascending/descending	Sorts table ascending/descending	Does not sort ascending/descending	Failed
Req 7: Navigation Bar			
Redirects user to correct page when clicking button	Redirects user to correct page when clicking button	Redirects user to correct page when clicking button	Passed

Code Inspection Document

Login Page

The layout of the login page was designed to follow a pleasant looking blue coloured theme, with the title of the system stated clearly at the top of the page, as well as a simple, easy to use login menu.

In order to not confuse the user, the login form is very straight forward and does not require much context to use:

```
<!-- Login Form -->
<form autocomplete="off" class="form" action="" method="POST">
<h2 class="form-header"> Enter your Details: </h2>
<br>

<div>
  <label class="label">User ID: </label>          <!-- 'User ID' input field, required field that only takes numbers -->
  <input class="form-input" type="number" name="userID" required placeholder="Unique 8 Digit Number">
</div>

<br>

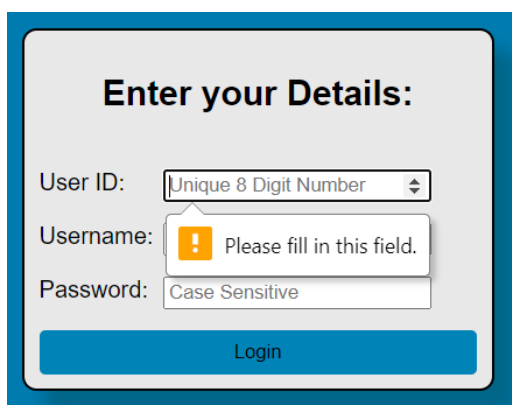
<div>
  <label class="label">Username: </label>          <!-- 'Username' input field, required field -->
  <input class="form-input" type="text" name="username" required placeholder="e.g. 'j.doe'">
</div>

<br>

<div>
  <label class="label">Password: </label>          <!-- 'Password' input field, required field -->
  <input class="form-input" type="password" name="password" required placeholder="Case Sensitive">
</div>

<br>
<br>
  <input class="submit-button" type="submit" value="Login">          <!-- 'Login' Button -->
</div>
</form>          <!-- End of Login form -->
```

All input fields in the login form are required and will show an alert when attempting to login without filling them all in:



When hovering over the 'Login' button, the colour will change:

```

/* 'Login' Button CSS */
.submit-button {
    display: block;
    width: 100%;
    background-color: #0083B7;
    border: none;
    border-radius: 4px;
    padding: 7px;
    cursor: pointer;
    text-align: center;
}

/* Change colour of 'Login' button when hovered over */
.submit-button:hover {
    background-color: #009DDB;
}

```

When all of the details entered are correct and the ‘Login’ button is clicked, the page will automatically redirect the user to their respective page based on their user type (if they are a student, they will be redirected to the student page, if they are a supervisor, they will be redirected to the supervisor page and so on):

```

//redirect to the correct page based on usertype
if($row["usertype"] == "student") {
    //if 'user' is a student
    $_SESSION["userID"] = $userID; //use their userID to begin a login session
    $id = $row['userID']; //use their userID to retrieve only the user
    header("location: http://localhost/enterpriseprro/student/home/studhome.php?id=".$id); //red
}

elseif($row["usertype"] == "supervisor") {
    //if 'user' is a supervisor
    $_SESSION["userID"] = $userID; //use their userID to begin a login session
    $id = $row['userID']; //use their userID to retrieve only the user
    header("location:http://localhost/enterpriseprro/supervisor/home/sprhome.php?id=".$id); //red
}

elseif($row["usertype"] == "module coordinator") {
    //if 'user' is a module coordinator
    $_SESSION["userID"] = $userID; //use their userID to begin a login session
    $id = $row['userID']; //use their userID to retrieve only the user
    header("location: http://localhost/enterpriseprro/module-coordinator/students/students.php?id=".$id); //red
}

elseif($row["usertype"] == null) {
    //if user's details (userID, username or password) are not correct
    header("location: http://localhost/enterpriseprro/login/login.php"); //do not allow access and
}

```

[Text Wrapping Break]

Module Coordinator – View Students Page

Following on from the login page, the layout of this page has the same blue theme with the title and subtitle of the system displayed clearly at the top.

This page contains a navigation bar, with links to the ‘View Supervisors’ Page, ‘View Projects’ Page and a button to log out:

```

<!-- Navigation bar Links -->
<div class="navbar">
  <a href="http://localhost/enterprise/module-coordinator/students/students.php">Students</a>
  <a href="http://localhost/enterprise/module-coordinator/supervisors/supervisors.php">Supervisors</a>
  <a href="http://localhost/enterprise/module-coordinator/projects/projects.php">Projects</a>
  <a href="http://localhost/enterprise/login/login.php">Logout</a>
</div>

```

There is also a feature that allows the user to add or update a student's information. By default, this form allows the user to add a new student, but by clicking the 'Update' button in the database table, the form changes to 'Update Student Information'. The form fetches information such as 'Supervisor ID' and 'Supervisor Name' directly from the database:

```

<!-- 'Add New Student' & 'Update Student Info' Form -->
<form autocomplete="off" class="form" method="POST" action="student-crud.php">
  <?php if ($update == false): ?>      <!-- Default state of form: when 'update' button is not clicked, by default the form will 'add' students -->
    <h2 class="form-header"> Add a New Student: </h2>      <!-- Title of form -->
  <?php else: ?>
    <!-- When 'update' button is clicked, form changes to update student information -->
    <h2 class="form-header"> Update Student Information: </h2>      <!-- Title of form -->
  <?php endif ?>

  <div class="form-input">
    <label>Student ID: </label>      <!-- 'Student ID' input field, required field that only takes numbers -->
    <input type="number" name="studentNo" required value="<?php echo $studentNo; ?>">
  </div>

  <div class="form-input">
    <label>First Name: </label>      <!-- 'First Name' input field, required field -->
    <input type="text" name="fName" required value="<?php echo $fName; ?>">
  </div>

  <div class="form-input">
    <label>Last Name: </label>      <!-- 'Last Name' input field, required field -->
    <input type="text" name="lName" required value="<?php echo $lName; ?>">
  </div>

  <div class="form-dropdown">
    <label>Supervisor ID: </label>      <!-- 'Supervisor ID' dropdown list -->
    <select id="supervisorNo" name="sprvisorNo" value="<?php echo $sprvisorNo; ?>">
      <option value="" disabled selected>Select</option>
      <?php $sql = "SELECT sprvisorNo FROM supervisors"; //fill dropdown list directly with database records
      $result = mysqli_query($connection, $sql);
      while ($row = mysqli_fetch_array($result)) {
        echo '<option>'.$row['sprvisorNo'].'</option>'; //display database records as dropdown list options
      } ?>
    </select>
  </div>

  <div class="form-sprvisor">
    <label>Supervisor: </label>      <!-- 'Supervisor' dropdown list -->
    <select id="supervisors" name="sprvisor" onChange="changeproj(this.value);" value="<?php echo $sprvisor; ?>"> <!-- Based on the supervisor selected from the list,
      <option value="" disabled selected>Select</option>
      <?php $sql = "SELECT sprvisorName FROM supervisors"; //fill dropdown list directly with database records
      $result = mysqli_query($connection, $sql);
      while ($row = mysqli_fetch_array($result)) {
        echo '<option>'.$row['sprvisorName'].'</option>'; //display database records as dropdown list options
      } ?>
    </select>
  </div>

  <div class="form-project">
    <label>Allocated Project: </label>      <!-- 'Allocated Project' dropdown list - based on supervisor selected, this list will only show their projects -->
    <select id="projects" name="project" value="<?php echo $project; ?>">
      <option value="" disabled selected>Select</option>
    </select>
  </div>

```

There is also a search bar function to search through the records in the database:

```

<!-- Search bar function -->
<form autocomplete="off" name="search_form" class="search-bar" method="POST" action="students.php">
  <input type="text" name="search_box" placeholder="Search..." value="" />      <!-- Search box text input field -->
  <input type="submit" name="search" class="search-button" value="Search Table"> <!-- 'Search Table' button -->
</form>                                <!-- End of Search bar function -->

//retrieves data that was searched in the Search Bar
$searchsql = "SELECT * FROM students ";
if (isset($_POST['search'])) {          //if 'Search Table' button is clicked
  $search_term = mysqli_real_escape_string($connection, $_POST['search_box']);

  //display records that contains the same characters as in the search term
  $searchsql .= "WHERE studentNo LIKE '%{$search_term}%'";
  $searchsql .= " OR fName LIKE '%{$search_term}%'";
  $searchsql .= " OR lName LIKE '%{$search_term}%'";
  $searchsql .= " OR project LIKE '%{$search_term}%'";
  $searchsql .= " OR sprvisor LIKE '%{$search_term}%'";
}
$results = mysqli_query($connection,$searchsql);

```

Finally, the database table shows a list of all of the students, along with their details and allocated project and supervisor. This information is retrieved from the database and is automatically updated when any changes are made or if a student is added or deleted:

```

//executes the correct columns from the SQL Database
$sql="SELECT studentNo, fName, lName, project, sprvisor FROM students";
$results=mysqli_query($connection,$sql);

```

```

//displays all records from database table
if (mysqli_num_rows($results)>0) {
  while($row=mysqli_fetch_array($results)) {
    echo "<tr>
      <td>" . $row["studentNo"] . "</td>          <!-- display student IDs -->
      <td>" . $row["fName"] . "</td>              <!-- display student first names -->
      <td>" . $row["lName"] . "</td>              <!-- display student last names -->
      <td>" . $row["project"] . "</td>            <!-- display student's allocated projects -->
      <td>" . $row["sprvisor"] . "</td>           <!-- display student's allocated supervisors -->
      <td>
        <a href='students.php?update=". $row["studentNo"] . "' class='update-btn'>Update</a>      <!-- 'Update' button -->
        <a href='student-crud.php?del=" . $row["studentNo"] . "' class='delete-btn'>Delete</a>    <!-- 'Delete' button -->
      </td>
    </tr>";
  }
}

```

To maintain standardisation and avoid any confusion, the same layout is established in the Module Coordinator ‘View Supervisors’ and ‘View Projects’ pages.

Student Home Page

The Student’s Home Page follows the same blue themed layout and has navigation bar links to the ‘Home’ page, ‘Choose Project’ page and to logout:

```

<!-- Navigation bar Links -->
<div class="navbar">
  <a href="http://localhost/enterprisepro/student/home/studhome.php">Home</a>
  <a href="http://localhost/enterprisepro/student/chooseproject/chooseproject.php">Choose Project</a>
  <a href="http://localhost/enterprisepro/login/login.php">Logout</a>
</div>

```

Students can use this page to view their own specific details and allocated project by retrieving their records based on their userID. The code does not allow a student to view any other student's details or to edit or delete them in any way:

```
//executes the correct columns from the SQL Database
$_SESSION["userID"];
$sql="SELECT studentNo, fName, lName, project, sprvisor
FROM students s INNER JOIN users u ON s.studentNo = u.userID
WHERE s.studentNo = '{$_SESSION['userID']}'"; //uses the student's userID to retrieve only their specific, personal information
$results=mysqli_query($connection,$sql);

//displays the user's specific, personal records from database table
if (mysqli_num_rows($results)>0) {
while($row=mysqli_fetch_array($results)) {
    echo "<tr>
    <td>" . $row["studentNo"] . "</td>           <!-- display user's studentID -->
    <td>" . $row["fName"] . "</td>               <!-- display user's first name -->
    <td>" . $row["lName"] . "</td>               <!-- display user's last name -->
    <td>" . $row["project"] . "</td>             <!-- display user's allocated project -->
    <td>" . $row["sprvisor"] . "</td>            <!-- display user's allocated supervisor -->
    </tr>";
}
}
```

Student's Choose Project Page

This page can be accessed directly from the student Home page and contains navigation bars to the other student pages:

```
<!-- Navigation bar Links -->
<div class="navbar">
    <a href="http://localhost/enterprisepro/student/home/studhome.php">Home</a>
    <a href="http://localhost/enterprisepro/student/chooseproject/chooseproject.php">Choose Project</a>
    <a href="http://localhost/enterprisepro/login/login.php">Logout</a>
</div>
```

On this page, students can view all of the available projects in a database table that automatically updates when projects are added, updated or deleted by supervisors or the module coordinator:

```
//executes the correct columns from the SQL Database
$_SESSION["userID"];
$sql="SELECT projectNo, projectName, projectDesc, sprvisor FROM projects";
$results=mysqli_query($connection,$sql);

//displays all records from database table
if (mysqli_num_rows($results)>0) {
while($row=mysqli_fetch_array($results)) {
    echo "<tr>
    <td>" . $row["projectNo"] . "</td>           <!-- display project IDs -->
    <td>" . $row["projectName"] . "</td>         <!-- display project Names -->
    <td><button type='button' class='collapsible'>View Description</button> <!-- 'View Description' Collapsible Box -->
    <div class='collapsible-text'>
    <p>" . $row["projectDesc"] . " </p> </div></td> <!-- display project Descriptions -->
    <td>" . $row["sprvisor"] . "</td>           <!-- display project supervisors -->
    </tr>";
}
}
```

To avoid making the table look messy and show too much unnecessary information at one time, the description of projects is stored in a collapsible box, that when clicked, opens to reveal the information:

```

<!-- JavaScript for 'View Description' collapsible box - When clicked, box shows project description -->
<script>
var coll = document.getElementsByClassName("collapsible");
var i;

for (i = 0; i < coll.length; i++) {
  coll[i].addEventListener("click", function() {
    this.classList.toggle("active");
    var content = this.nextElementSibling;
    if (content.style.display === "block") {
      content.style.display = "none";
    } else {
      content.style.display = "block";
    }
  });
}
</script>

```

The page also has a search bar function to easily find database records:

```

<!-- Search bar function -->
<form autocomplete="off" name="search_form" class="search-bar" method="POST" action="chooseproject.php">
  <input type="text" name="search_box" placeholder="Search..." value="" /> <!-- Search box text input field -->
  <input type="submit" name="search" class="search-button" value="Search Table"> <!-- 'Search Table' Button -->
</form> <!-- End of Search Bar Function -->

//retrieves data that was searched
$searchsql = "SELECT * FROM projects ";
if (isset($_POST['search'])) { //if 'Search Table' button is clicked
  $search_term = mysqli_real_escape_string($connection, $_POST['search_box']);

  //display records that contains the same characters as in the search term
  $searchsql .= "WHERE projectNo LIKE '%{$search_term}%'";
  $searchsql .= " OR projectName LIKE '%{$search_term}%'";
  $searchsql .= " OR sprvisor LIKE '%{$search_term}%'";
}
$results = mysqli_query($connection,$searchsql);

```

The student can also update their chosen project using the form below:

```

<!-- Update Project Form -->
<form autocomplete="off" class="form" method="POST" action="chooseprojectcrud.php">
<h2 class="form-header"> Select from the list: </h2> <!-- Form Instruction -->

<div class="form-input">
  <label>Supervisor: </label> <!-- 'Supervisor' dropdown list -->
  <select id="supervisors" name="sprvisor" onChange="changeproj(this.value);" value="<?php echo $sprvisor; ?>">
    <option value="" disabled selected>Select</option>
    <?php $sql = "SELECT sprvisorName FROM supervisors"; //fill dropdown list directly with database records
    $result = mysqli_query($connection, $sql);
    while ($row = mysqli_fetch_array($result)) {
      echo '<option>'.$row['sprvisorName'].'</option>'; //display database records as dropdown list options
    } ?>
  </select>
</div>

<div class="form-input">
  <label>Allocated Project: </label> <!-- 'Allocated Project' dropdown list - based on supervisor's projects -->
  <select id="projects" name="project" value="<?php echo $project; ?>">
    <option value="" disabled selected>Select</option>
  </select>

```

After updating their chosen project, a message is displayed to the screen:

```

<!-- Alert Message when updating project -->
<?php if (isset($_SESSION['msg'])): ?>
    <div class="msg">
        <?php
            echo $_SESSION['msg'];
            unset($_SESSION['msg']);
        ?>
    </div>
<?php endif ?>

```

Supervisor Home Page

The supervisor home page contains navigation bar links to the supervisor pages – the ‘Home’ page, the ‘View Allocated Students’ page and to logout:

```

<!-- Navigation bar Links -->
<div class="navbar">
    <a href="http://localhost/enterprisepr/supervisor/home/sprhome.php">Home</a>
    <a href="http://localhost/enterprisepr/supervisor/viewstudents/viewstudents.php">View Allocated Students</a>
    <a href="http://localhost/enterprisepr/login/login.php">Logout</a>
</div>

```

Based on the supervisor’s userID, the supervisor can only view their own details and will not be able to view anyone else’s information or edit or delete them:

```

//executes the correct columns from the SQL Database
$_SESSION["userID"];
$sql="SELECT sprvisorNo, fName, lName
FROM supervisors a INNER JOIN users b ON a.sprvisorNo = b.userID
WHERE a.sprvisorNo = '{$_SESSION['userID']}'"; //uses the supervisor's userID to retrieve only their specific, personal information
$results=mysqli_query($connection,$sql);

//displays the user's specific, personal records from database table
if (mysqli_num_rows($results)>0) {
    while($row=mysqli_fetch_array($results)) {
        echo "<tr>
            <td> . $row['sprvisorNo'] . "</td>           <!-- displays user's supervisorID -->
            <td> . $row['fName'] . "</td>               <!-- displays user's first name -->
            <td> . $row['lName'] . "</td>               <!-- display user's last name -->
        </tr>";
    }
}

```

They are also able to view their own projects:

```

//executes the correct columns from the SQL Database
$_SESSION["userID"];
$sql="SELECT projectNo, projectName, projectDesc
FROM projects a INNER JOIN users b ON a.sprvisorNo = b.userID
WHERE a.sprvisorNo = '{$_SESSION['userID']}'"; //uses the supervisor's userID to retrieve only their specific projects
$results=mysqli_query($connection,$sql);

//displays the user's specific project records from database table
if (mysqli_num_rows($results)>0) {
    while($row=mysqli_fetch_array($results)) {
        echo "<tr>
            <td> . $row['projectNo'] . "</td>           <!-- display project IDs -->
            <td> . $row['projectName'] . "</td>         <!-- display project names -->
            <td><button type='button' class='collapsible'>View Description</button> <!-- 'View Description' Collapsible Box -->
            <div class='collapsible-text'>
                <p> . $row['projectDesc'] . " </p> </div></td> <!-- display project descriptions -->
            <td>
                <a href='sprhome.php?update=" . $row['projectNo'] . "' class='update-btn'>Update</a> <!-- 'Update' Button -->
                <a href='sprhome.php?del=" . $row['projectNo'] . "' class='delete-btn'>Delete</a> <!-- 'Delete' Button -->
            </td>
        </tr>";
    }
}
} ?>

```


Supervisors can also add, update or delete their own projects by filling out the form. By default the form allows the user to first add a project, but if they click the ‘Update’ button in the database table, the form will automatically change to ‘Update Project Information’:

```
<!-- 'Add New Project' & 'Update Project Info' Form -->
<form autocomplete="off" class="form" method="POST" action="sprhomecrud.php">
  <?php if ($update == false): ?>
    <h2 class="form-header"> Add a New Project: </h2>
    <!-- Default state of form: when 'update' button is not clicked, by default the form changes to update project information -->
  <?php else: ?>
    <h2 class="form-header"> Update Project Information: </h2>
    <!-- When 'update' button is clicked, form changes to update project information -->
  <?php endif ?>

  <div class="form-input">
    <label>Project ID: </label>
    <input type="text" name="projectNo" required value="<?php echo $projectNo; ?>" />
    <!-- 'Project ID' input field, required field that only takes numbers -->
  </div>

  <div class="form-input">
    <label>Project Name: </label>
    <input type="text" name="projectName" required value="<?php echo $projectName; ?>" />
    <!-- 'Project Name' input field, required field -->
  </div>

  <div class="form-input">
    <label>Project Description: </label>
    <input type="text" name="projectDesc" required value="<?php echo $projectDesc; ?>" />
    <!-- 'Project Description' input field, required field -->
  </div>

  <div class="form-input">
    <label>Supervisor ID: </label>
    <select id="supervisorsID" name="sprvisorNo" value="<?php echo $sprvisorNo; ?>">
      <?php $sql = "SELECT sprvisorNo FROM supervisors
        WHERE sprvisorNo = '{$_SESSION['userID']}';"
        $result = mysqli_query($connection, $sql);
        while ($row = mysqli_fetch_array($result)) {
          echo '<option>'.$row['sprvisorNo'].'</option>';
        }
      </select>
    <!-- 'Supervisor ID' dropdown list -->
    <!-- supervisor can only add/update their own projects, not anyone else's -->
    <!-- display only the logged in supervisor's supervisorID as the dropdown list option -->
  </div>

  <div class="form-input">
    <label>Supervisor: </label>
    <select id="supervisors" name="sprvisor" value="<?php echo $sprvisor; ?>">
      <?php $sql = "SELECT sprvisorName FROM supervisors
        WHERE sprvisorNo = '{$_SESSION['userID']}';"
        $result = mysqli_query($connection, $sql);
        while ($row = mysqli_fetch_array($result)) {
          echo '<option>'.$row['sprvisorName'].'</option>';
        }
      </select>
    <!-- 'Supervisor' dropdown list -->
    <!-- supervisor can only add/update their own projects, not anyone else's -->
    <!-- display only the logged in supervisor's name as the dropdown list option -->
  </div>

  <div class="form-input">
    <?php if ($update == false): ?>
      <button type="submit" name="add" class="form-btn">Add Project</button>
    <?php else: ?>
      <button type="submit" name="update" class="form-btn">Update Record</button>
    <?php endif ?>
    <!-- Default state of form, when 'update' button is not clicked, by default the form changes to show 'Update Record' button -->
    <!-- 'Add Project' Button -->
    <!-- 'Update Record' Button -->
  </div>
</form>
<!-- End of 'Add Project'/'Update Project Information' Form -->
```

When a project is added, updated or deleted, a confirmation message will appear on the screen:

```
<!-- Message displayed when records added, updated or deleted -->
<?php if (isset($_SESSION['msg'])): ?>
  <div class="msg">
    <?php
      echo $_SESSION['msg'];
      unset($_SESSION['msg']);
    ?>
  </div>
<?php endif ?>
<br>
```

Supervisor View Allocated Students Page

This page contains navigation bar links to the other supervisor pages – the ‘Home’ page, the ‘View Allocated Students’ page and to logout:

```
<!-- Navigation bar Links -->
<div class="navbar">
  <a href="http://localhost/enterprisepro/supervisor/home/sprhome.php">Home</a>
  <a href="http://localhost/enterprisepro/supervisor/viewstudents/viewstudents.php">View Allocated Students</a>
  <a href="http://localhost/enterprisepro/login/login.php">Logout</a>
</div>
```

On this page, supervisors can view their own allocated students who have chosen one of their projects. The database table uses the supervisor’s userID to retrieve only their specific allocated students and displays their information and allocated project:

```
//executes the correct columns from the SQL Database
$_SESSION["userID"];
$sql="SELECT st.studentNo, st.fName, st.lName, st.project
FROM students st INNER JOIN supervisors su ON st.sprvisorNo = su.sprvisorNo
WHERE su.sprvisorNo = {$_SESSION['userID']} "; //uses the supervisor's userID to retrieve
$results=mysqli_query($connection,$sql);

//displays the user's specific, allocated students from database table
if (mysqli_num_rows($results)>0) {
while($row=mysqli_fetch_array($results)) {
  echo "<tr>
    <td>" . $row["studentNo"] . "</td>           <!-- displays student IDs -->
    <td>" . $row["fName"] . "</td>               <!-- displays student first names -->
    <td>" . $row["lName"] . "</td>               <!-- displays student last names -->
    <td>" . $row["project"] . "</td>             <!-- display student projects -->
  </tr>";
}
}
```