

Technical Manual

MYSQL and PHP

Question 1

```
DROP FUNCTION IF EXISTS CheckPassword;
DELIMITER $$
CREATE FUNCTION CheckPassword (username VARCHAR(10), userpassword VARCHAR(15))
RETURNS BOOL
NOT DETERMINISTIC
READS SQL DATA
BEGIN
RETURN EXISTS (SELECT PoliceUser_username FROM PoliceUser WHERE PoliceUser_username = username AND PoliceUser_password =
userpassword);
END;
$$
DELIMITER ;
```

This Stored Function returns a 1 if the username and password are exactly correct and 0 if they are not. If a one is outputted then a user or admin will be able to continue to the next page

```
DROP FUNCTION IF EXISTS CheckAdmin;
DELIMITER $$
CREATE FUNCTION CheckAdmin (username VARCHAR(10), userpassword VARCHAR(15))
RETURNS BOOL
NOT DETERMINISTIC
READS SQL DATA
BEGIN
RETURN EXISTS (SELECT PoliceUser_admin FROM PoliceUser WHERE PoliceUser_username =
username AND PoliceUser_password = userpassword);
END;
$$
DELIMITER ;
SELECT CheckAdmin('Daniels','copper99');
```

```
Username: &nbsp;<input type="text" name="Username"
Id="Username" required><br>
Password: <input type="text" name="Password" placeholder="Password"
d > <br>
Admin: <select name="Admin" id="Admin">
<option value=1>Admin</option>
<option value=0 selected>User </option>
</select><br>
<input type="submit" name ="Enter" value = "Enter" required>
```

```
// If the enter button is clicked
if(isset($_POST['Enter'])){

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

    $Policeusername = trim($_POST['Username']);
    $Policepassword = trim($_POST['Password']);
    $Policeadmin = trim($_POST['Admin']);

// Perform query
$result = $mysqli -> query("SELECT
CheckPassword('$Policeusername','$Policepassword')"); {
#echo "Returned rows are: " . $result -> num_rows;
$row = $result -> fetch_row() ;

// Free result set
if ($row[0] == 0 && $Policeadmin == 0){
    $_SESSION["username"] = $Policeusername;
    $_SESSION["Admin"] = $Policeadmin;
    header("location:Home.php");
}
if ($row[0] == 0 && $Policeadmin == 1){

    echo "<script>alert('You do not have Admin permissions');</script>";

}
if ($row[0] == 1 && $Policeadmin == 1 ){
    $_SESSION["username"] = $Policeusername;
    $_SESSION["Admin"] = $Policeadmin;
    header("location:Home.php");
}
if ($row[0] == 1 && $Policeadmin == 0){
    echo "<script>alert('Try logging onto Admin buddy');</script>";
}
else if ($row[0] == -1) {
    echo "<script>alert('Wrong password');</script>";
}
```

The stored function for the admin does the same thing as the password. If the username is correct and the admin is correct then they will be able to log in with admin features else if the username is correct and the admin isn't correct then they can log in with normal police features

```

DROP FUNCTION IF EXISTS ChangePassword;
Delimiter $$
CREATE FUNCTION ChangePassword(username VARCHAR(10),Oldpassword VARCHAR(15),Newpassword
VARCHAR(15),NewpasswordCheck VARCHAR(15))
RETURNS INT
NOT DETERMINISTIC
READS SQL DATA
BEGIN
    IF Newpassword <> NewpasswordCheck THEN
        Return 1; # Confirm New password, error message for passwords do not match
    END IF;
    IF EXISTS(SELECT * FROM PoliceUser WHERE PoliceUser_username = username AND
PoliceUser_password = Oldpassword) THEN
        UPDATE PoliceUser SET PoliceUser_password = Newpassword WHERE PoliceUser_username =
username;
        Return 0; # Password changed
    else
        RETURN 2; # Password does not match old password so do an error message
    END IF;
END;
$$
DELIMITER ;
SELECT ChangePassword('psxng3','psxng3','psxng3','psxng3');

```

Done

☰

ChangePassword (10 of 39)

⬆

```

}

// If the enter button is clicked
if(isset($_POST['Enter'])) {

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

    $Policeusername = $_SESSION["username"];
    $Policepassword = trim($_POST['Password']);
    $NewPassword = trim($_POST['NewPassword']);
    $ConfirmPassword = trim($_POST['ConfirmPassword']);

    // Store Query in result
    $result = $mysqli->query("SELECT
ChangePassword('$Policeusername','$Policepassword','$NewPassword','$ConfirmPass
word')"); {
        //echo "Returned rows are: " . $result->num_rows;
        $row = $result->fetch_row();
        //echo $row[0];
        // Free result set
        if ($row[0] == 0) {
            echo "<p>Password Changed</p>";
        }
        else if ($row[0] == 1) {
            echo "<p>New Passwords do not match!</p>";
        }
        else if ($row[0] == 2) {
            echo "<p>Password does not match old password</p>";
        }
        // Frees result memory
        $result->free_result();
    }

    $mysqli->close();
}

```

Return ps integer values of 0,1 and 2. If the new password inputted is not equal to the confirm new password then return 0 and a "password is not the same password error message. If old password is not equal to current password in the database then return 2 and error message for the password not matching. When all the correct parameters are put in then the password will be set to be the same as the password we set in new password and confirm new password.

Question 2

```
DROP PROCEDURE IF EXISTS sp_CheckNameCheckDLSearch;
DELIMITER $$
CREATE PROCEDURE sp_CheckNameCheckDLSearch(IN firstname VARCHAR(50),IN lastname
VARCHAR(50), IN licence VARCHAR(16))
BEGIN
IF (firstname <> '' AND lastname = '') THEN
    SELECT * FROM People WHERE (People_Firstname LIKE firstname) ; # Confirm New password,
error message for passwords do not match
ELSEIF (firstname = '' AND lastname <> '') THEN
    SELECT * FROM People WHERE (People_Lastname LIKE lastname) ;
ELSEIF (firstname <> '' AND lastname <> '' THEN
    SELECT * FROM People WHERE (People_Firstname LIKE firstname) AND (People_Lastname LIKE
lastname);
ELSEIF licence <> '' THEN
    SELECT * FROM People WHERE People_licence = licence ;
END IF;
```

```
END $$
DELIMITER;
```

```
if(isset($_POST['Enter'])) {

    // Setting variables = to input in the forms
    $Firstname = trim($_POST['Firstname']);
    $Lastname = trim($_POST['Lastname']);
    $DL = trim($_POST['Driving']);

    $result = $mysqli -> query("CALL
sp_CheckNameCheckDLSearch('$Firstname','$Lastname','$DL')");{
```

```

echo "<table border='10'>

<tr>

<th>People ID</th>

<th>First Name</th>

<th>Last Name</th>

<th>Address</th>

<th>Driving Licence</th>

</tr>";

//$col = array();
while($row = $result -> fetch_row() )
{

    //$row = $result -> fetch_row() ;
    //$row = mysqli_fetch_array($result);

    "<tr>";
    echo "<td>".$row[0]."</td>";
    echo "<td>".$row[1]."</td>";
    echo "<td>".$row[2]."</td>";
    echo "<td>".$row[3]."</td>";
    echo "<td>".$row[4]."</td>";
    echo "</tr>";
}

```

Stored procedure – if 1st name is not equal to empty string and last name is then carry out a search for first name else if the other way around do an empty string for last name. Else if they both have parameters then search for both first name and last name. When licence is not empty, then we can search with licence. This should bring up the people tables with the information included matching the details of the input Ed first name, last name and driving licence

Question 3

```
DROP PROCEDURE IF EXISTS sp_VehicleLicenceSearch;
DELIMITER $$
CREATE PROCEDURE sp_VehicleLicenceSearch(IN vehicleLicence
VARCHAR(7))
BEGIN
SELECT People_Firstname AS 'First Name', People_Lastname AS 'Last
Name', People_licence AS 'Driving Licence', Vehicle_type AS 'Vehicle Type',
Vehicle_colour AS 'Vehicle Colour'
FROM Vehicle LEFT JOIN Incident ON Vehicle.Vehicle_ID =
Incident.Vehicle_ID LEFT JOIN People ON People.People_ID =
Incident.People_ID/*LEFT JOIN Incident USING(Vehicle_ID) = (Incident_ID)*/
WHERE Vehicle_licence = vehicleLicence;

END $$
DELIMITER;
```

The bit after Incident.people_ID is commented out
LEFT JOIN is used to allow null values. Vehicle ID on incident ID and

The table is loaded the same way as the table on Question 2

Question 4

```
DROP FUNCTION IF EXISTS AddNewPerson;
Delimiter $$
CREATE FUNCTION AddNewPerson(fname VARCHAR(50),lname VARCHAR(50),address VARCHAR(50),licence VARCHAR(16))
RETURNS Bool
NOT DETERMINISTIC
READS SQL DATA
```

```
BEGIN
```

```
    declare newid int;
```

```
    declare exit handler for sqlexception
```

```
    begin
```

```
        return 0;
```

```
    end;
```

```
    INSERT INTO People (People_Firstname,People_Lastname,People_address, People_licence) VALUES
    (fname ,lname , address , licence);
```

```
    SELECT LAST_INSERT_ID() into newid;
```

```
    return newid;
```

```
END;
```

```
$$
```

```
DROP PROCEDURE IF EXISTS sp_VehicleLicenceSearch;
```

```
DELIMITER $$
```

```
CREATE PROCEDURE sp_VehicleLicenceSearch(IN vehicleLicence
VARCHAR(7))
```

```
BEGIN
```

```
SELECT People_Firstname AS 'First Name', People_Lastname AS 'Last
Name', People_licence AS 'Driving Licence', Vehicle_type AS 'Vehicle Type',
Vehicle_colour AS 'Vehicle Colour'
```

```
FROM Vehicle LEFT JOIN Incident ON Vehicle.Vehicle_ID =
Incident.Vehicle_ID LEFT JOIN People ON People.People_ID =
Incident.People_ID/*LEFT JOIN Incident USING(Vehicle_ID) = (Incident_ID)*/
```

```
WHERE Vehicle_licence = vehicleLicence;
```

```
END $$
```

```
DELIMITER;
```



```

DROP FUNCTION IF EXISTS AddNewVehicle;
Delimiter $$
CREATE FUNCTION AddNewVehicle(Vlicence VARCHAR(7),Vtype
VARCHAR(20),Vcolour VARCHAR(20),PeopleID INT(11))
RETURNS Bool
NOT DETERMINISTIC
READS SQL DATA

BEGIN

    declare newid int;

    declare exit handler for sqlexception
    begin
        return 0;
    end;

    INSERT INTO Vehicle (Vehicle_licence, Vehicle_type, Vehicle_colour )
VALUES
    (Vlicence ,Vtype ,Vcolour);

    SELECT LAST_INSERT_ID() into newid;
    INSERT INTO Ownership(People_ID, Vehicle_ID) VALUES
(PeopleID,newid);

    return newid;

END;
$$
#DELIMITER ;
SELECT AddNewVehicle('TRZ2084','Mercedes Benz','Grey',4);

```

Question 5

```
DROP FUNCTION IF EXISTS CreateIncidentReport;
Delimiter $$
CREATE FUNCTION CreateIncidentReport(VehicleID VARCHAR(11),PeopleID VARCHAR(11),IncidentDate VARCHAR(10),IncidentReport
VARCHAR(500),OffenceID VARCHAR(11))
RETURNS Bool
NOT DETERMINISTIC
READS SQL DATA
```

```
BEGIN
```

```
    declare newid int;
```

```
    declare exit handler for sqlexception
```

```
    begin
```

```
        return 0;
```

```
    end;
```

```
    INSERT INTO Incident (Vehicle_ID,People_ID,Incident_Date, Incident_Report, Offence_ID) VALUES
    (VehicleID ,PeopleID ,IncidentDate ,IncidentReport ,OffenceID );
    return Incident_ID;
```

```
END;
```

```
$$
```

```
DROP PROCEDURE IF EXISTS sp_SearchIncident;
```

```
DELIMITER $$
```

```
CREATE PROCEDURE sp_SearchIncident(IN IncidentID INT(11),IN OffenceID INT(11))
```

```
BEGIN
```

```
SELECT Incident_ID, Vehicle_ID,People_ID, Offence_ID FROM Incident WHERE Incident_ID = IncidentID or Offence_ID = OffenceID;
```

```
END $$
```

```
DELIMITER;
```

12:00 Fri 24 Dec

...

25%

Done

selectIncident (31 of 35)



```
// Check connection
$mysqli = new mysqli($servername,$username,$password,$dbname);

// Check connection
if ($mysqli->connect_errno) {
echo "Failed to connect to MySQL: " . $mysqli->connect_error;
exit();
}

$ID = $_GET['ID'];

$result = $mysqli->query("CALL sp_SearchIncident('$IncidentID','$OffenceID')");
//echo "Returned rows are: " . $result->num_rows;
//echo "Returned rows are: " . $result2->num_rows;
//$row = $result->fetch_row();

$NumRows = $result->num_rows;
$Idrows = array();
$NameRows = array();

while($row = $result->fetch_row() )
{

// $row = $result->fetch_row();
// $row = mysqli_fetch_array($result);
```

12:00 Fri 24 Dec

...

25%

Done

ReportOptions (30 of 35)



```
<?php
// MySQL database information
$servername = "mysql.cs.nott.ac.uk";
$username = "psxng3";
$password = "NJJN1L";
$dbname = "psxng3";

// Check connection
$mysqli = new mysqli($servername,$username,$password,$dbname);

// Check connection
if ($mysqli->connect_errno) {
echo "Failed to connect to MySQL: " . $mysqli->connect_error;
exit();
}

// $Firstname = trim($_POST['Firstname']);
// $Lastname = trim($_POST['Lastname']);
// $DL = trim($_POST['Driving']);

// $result = $mysqli->query("CALL sp_CheckNameCheckDLSearch('$Firstname','$Lastname','$DL')");{
// echo "Returned rows are: " . $result->num_rows;
// echo "Returned rows are: " . $result2->num_rows;
// $row = $result->fetch_row();

// $rowcount = $result->num_rows;

// echo $rowcount;

/* echo "<table border='10'>

<tr>

<th>First Name</th>
```



```
<?php
// MySQL database information
$servername = "mysql.cs.nott.ac.uk";
$username = "psxng3";
$password = "NJJNIL";
$dbname = "psxng3";

// Check connection
mysqli = new mysqli($servername,$username,$password,$dbname);

// Check connection
if ($mysqli->connect_errno) {
echo "Failed to connect to MySQL: " . $mysqli->connect_error;
exit();
}

$VehicleID = $_GET['VehicleID'];
$PeopleID = $_GET['PeopleID'];
$IncidentDate = $_GET['IncidentDate'];
$OffenceID = $_GET['OffenceID'];

$result = $mysqli->query("SELECT
CreateIncidentReport('$VehicleID' , '$PeopleID' , '$IncidentDate' , '$IncidentReport' , '$OffenceID')");
//echo "Returned rows are: " . $result->num_rows;
//echo "Returned rows are: " . $result2->num_rows;
$row = $result->fetch_row();
```

Question 6

```
DROP FUNCTION IF EXISTS AddPoliceUser;
Delimiter $$
CREATE FUNCTION AddPoliceUser(username VARCHAR(10),fname VARCHAR(50),lname
VARCHAR(50),address VARCHAR(50),newpassword VARCHAR(15),admin TINYINT(1))
RETURNS Bool
NOT DETERMINISTIC
READS SQL DATA

BEGIN

    declare exit handler for sqlexception
    begin
        return 0;
    end;

    INSERT INTO PoliceUser (PoliceUser_username, PoliceUser_Firstname, PoliceUser_Lastname,
PoliceUser_address, PoliceUser_password, PoliceUser_admin ) VALUES
    (username , fname ,lname , address ,newpassword ,admin );
    return 1;

END;
$$
#DELIMITER ;
SELECT AddPoliceUser('MrsGhost','Angela','Valdez','23 Midlands Cottage,Nottingham,United
Kingdom','password','0');
```

Admin uses TinyINT to express it is a boolean

```

DROP FUNCTION IF EXISTS AddFines;
Delimiter $$
CREATE FUNCTION AddFines(FineAmount INT(11),FinePoints INT(11), IncidentID INT(11))
RETURNS int(11)
NOT DETERMINISTIC
READS SQL DATA
BEGIN
    declare offence_id int(11);
    declare maxFine int(11);
    declare maxPoints int(11);

    declare exit handler for sqlexception
    begin
        return 0;
    end;

    SELECT Offence_ID into @offence_id FROM Incident WHERE Incident_ID = 2;

    return offence_id;

    SELECT Offence_maxFine,Offence_maxPoints into maxFine, maxPoints FROM Offence WHERE
    Offence_ID = offence_id;

    IF FineAmount > maxFine THEN
        return -1;
    END IF;

    IF FinePoints > maxPoints THEN
        return -2;
    END IF;

    INSERT INTO Fines (Fine_Amount, Fine_Points,Incident_ID) VALUES
    (FineAmount,FinePoints,IncidentID);
    return 1;

END;
$$

SELECT AddFines (5000,6000,2)

```

Navbar

Session_start() is used to maintain the user session when using the website so every web page can tailored that unique individual.

In the index.html, the username and Admin boolean is stored in \$_session['username'] and \$_session['Admin']. I use a check on \$_session['Admin']] for if the admin is equal to 0 then it can only have user access while if admin is equal to 1 then more content will be displayed on it Navbar such as Fines and Add Police user.

```

<?php
if ($_SESSION['Admin'] == 0){
    echo "<li><a href='Home.php'>Home</a></li> <br><br>";
    echo "<li><a href='VehicleL_SearchPage.php'>VL
Search</a></li><br><br>";
    echo "<li><a href='Name_DL_SearchPage.php'>N DL
Search</a></li><br><br>";
    echo "<li><a href='VehicleOptions.php'>New Vehicle
Details</a></li><br><br>";
    echo "<li><a href='ReportOptions.php'>File Report </
a></li><br><br>";
    echo "<li><a href='ChangePassword.php'>Change
Password </a></li><br><br>";
}else
    if($_SESSION['Admin'] == 1){
        echo "<li><a href='Home.php'>Home</a></li><br>";
        echo "<li><a href='VehicleL_SearchPage.php'>VL Search</
a></li><br><br>";
        echo "<li><a href='Name_DL_SearchPage.php'>N DL
Search</a></li><br><br>";
        echo "<li><a href='VehicleOptions.php'>New Vehicle
Details</a></li><br><br>";
        echo "<li><a href='ReportOptions.php'>File Report </
a></li><br><br>";
        echo "<li><a href='ChangePassword.php'>Change Password
</a></li><br><br>";
        echo "<li><a href='ReportOptions.php'>Add Police User
</a></li><br><br>";
        echo "<li><a href='Fines.php'>Add Fines </a></
li><br><br>";
        echo "<li><a href='index.php'>Log Off </a></
li><br><br>";
    }
?>
</ul>
</nav>

```

Extra Functionality.

I have added a feature to allow a Driving License photo to be uploaded and stored in People MySQL table. A button is provided will display an upload control. When the form is submitted, the image file is read into memory and then converted into text using Base64 encoding. This is then stored in a LongText MySQL table column. When displaying the record, this field can be directly read and displayed on the form without the need to decode it, by using the img html tag.

The following PHP uploads the image file converts it to PHP and saves it to MySQL:

```

if(isset($_FILES['image'])){

```

```

echo "Image Saved to Database";
$errors= array();
$file_name = $_FILES['image']['name'];
$file_size = $_FILES['image']['size'];
$file_tmp = $_FILES['image']['tmp_name'];
$file_type=$_FILES['image']['type'];
$file_ext=strtolower(end(explode('.',$_FILES['image']['name'])));

$extensions= array("jpeg","jpg","png");

if(in_array($file_ext,$extensions)=== false){
    $errors[]="extension not allowed, please choose a JPEG or PNG file.";
}

$data = file_get_contents($_FILES['image']['name']);
$data = base64_encode($data);

$RecordID = $_COOKIE["imageid"];

$result = $mysqli->query("UPDATE People SET Image = '$data' WHERE People_ID = '$RecordID'");{

}

}

```

The application makes extensive use of JavaScript and JQuery. These development frameworks offer the following benefits:

- Can help in providing my responsive and dynamic web pages as they can interact with page elements without the need to submit and post, and reload the page. This improve the Web UI user experience. Some examples from the code are shown below:
- - **var firstname = \$("#Fname").val()** : Assigns the web Input Field Fname to the JavaScript variable firstname
 - **\$("#OwnerButton").hide()** : Hides the web button with id OwnerButton
 - **\$('#OffenceSelect > option').remove()**: Clears all the list of options in the OffenceSelect web dropdown control.
 - **\$("#Fname").val("")**: Clears the contexts of the Web Input Text field with the id Fname.

- It can be used for validating form entry before the form is submitted to the server. For example to validate the Input Text field with the id FName. The following code can be used:

```
if ( $("#Fname").val() == "")
{
    alert ("Please enter the First name before selecting this option")
    return
}
```

- It can be used to make Ajax calls which allow data to be sent to or retrieved from the without the need to submit and post, and reload a form. Ajax has been used extensively to save and read data from MySQL. The following JavaScript function is used save a vehicle entry into MySQL:

```
function saveNewVehicle ()
{

$.ajax({
    method: 'get',
    url: 'AddVehicle.php',
    data: {
        "Vlicence": $("#VL").val(),
        "Vtype": $("#VT").val(),
        "Vcolour" : $("#VC").val(),
        "PeopleID" : currentOwnerId

    },
    success: function(result) {

        var jsonResult = $.parseJSON(result);
        //currentOwnerId = jsonResult.Id;

        if (jsonResult.Success)
        {
            alert ("You have Successfully Created a Vehicle");

        }

        else
        {
            alert ("A problem Occurred while creating a Vehicle")
        }

        $("#addVechicleDialog").dialog("close");

    }
});
```

```
}
```

The are too many examples to list in this manual. There are comments in the code.