COMP1661 Logbook Upload Template

1.   Basic Information

1.1	Student name	**Myles Findlay**
1.2	Who did you work with? Note that for logbook exercises you are allowed to work with one other person as long as you give their name and login id and both contribute to the work.	**Name:** **Login id:**
1.3	Which Exercise is this? Tick as appropriate.	1 - Create a PhoneGap App utilizing the Notification API ☐ 2 - Create a PhoneGap data entry screen RentalZ App ☐ 3 - Create an SQLite database for RentalZ App ☒ 4 - Create Android data entry screen for RentalZ App ☐
1.4	How well did you complete the exercise? Tick as appropriate.	• I tried but couldn't complete it ☐ • I did it but I feel I should have done better ☐ • I did everything that was asked ☐ • I did more than was asked for ☒
1.5	Briefly explain your answer to question 1.4 e.g. what you could have done better?	n/a

2.   Exercise answer

2.1   Screenshots demonstrating what you achieved

This is the first screen the user will see when the app is launched. The user will be shown a greeting message along with a group of options that can be tweaked, incl. allowing the user to add a property, and showing all data with the database.
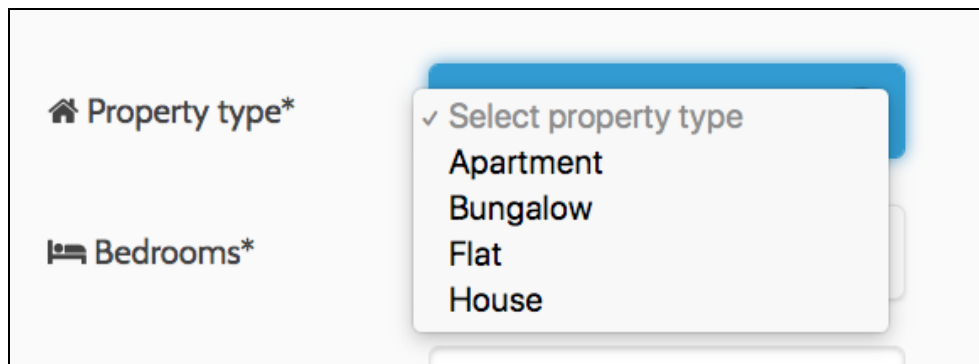
Figure 1: RentalZ form

Figure 2: 'Property type' select list

When the user taps the select list for 'Property type', they will be presented with a list of different kinds of properties to choose from (one must be chosen).
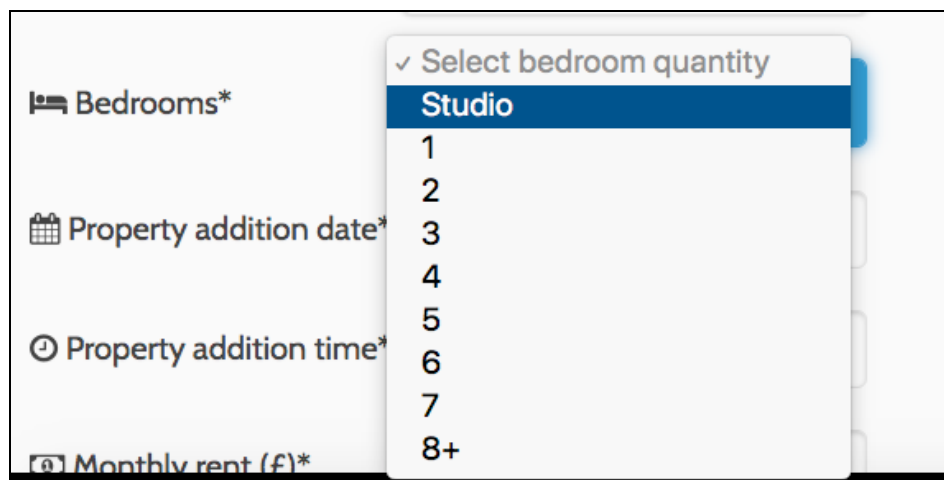

Figure 3: 'Bedrooms' select list

When the user taps the select list for 'Bedrooms', they will be presented with a list of numbers to choose from relating to the amount of bedrooms they want to specify for the property (one must be chosen).
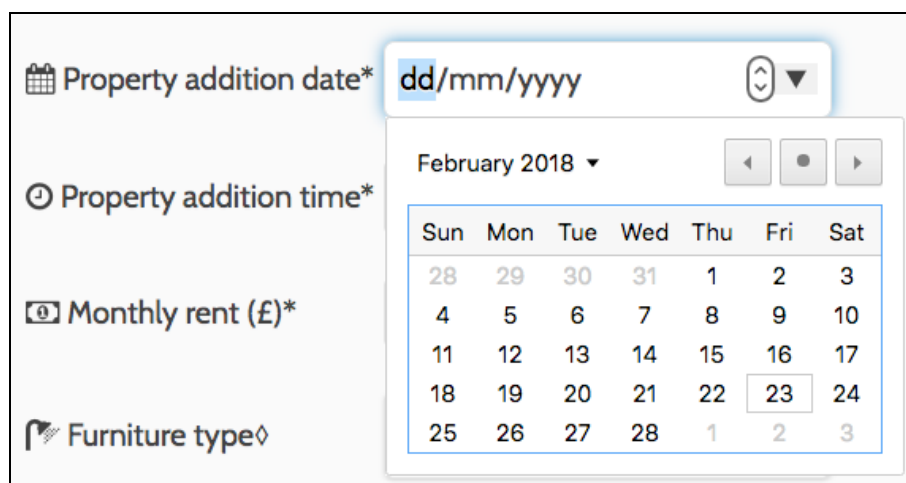

Figure 4: Date picker

When the user taps the input field for 'Property addition date', they will be presented with a DatePicker, which will allow them to choose a date for the property (a date must be chosen).

Figure 5: Time picker

When the user clicks the input field for 'Property addition time', they will be presented with a TimePicker, which will allow them to choose a time for the property (a time must be chosen).

Figure 5: 'Monthly rent' input field selected with 'number' type keyboard open

When the user clicks the input field for 'Monthly rent', they will be presented with a numeric keypad, which will allow them to choose a monetary value for the property (a numeric value must be specified). Thanks to `min="100"` in `<input type="number" min="100" id="monthlyRent" required>` 100 is the minimum value that must be specified.

Figure 6: 'Furniture type' select list

When the user clicks the select list for 'Furniture type', they will be presented with a list of different kinds of furnishing to choose from (this is optional).

Figure 7: 'Description' input field selected

An input field for 'Description' is also available, which will allow them add a description of the property (this is optional).

Figure 8: 'Reporter Name' input field selected with 'text' type keyboard open

The input field for 'Reporter name' contains a placeholder (Name Surname) informing the user what kind of information needs to be added to that field. I.e. a reporter's name and surname (this is required).

Figure 9: User-defined data (prior to being added to the database)

ID	Property Type	Bedrooms	Property Addition Date	Property Addition Time	Monthly Rent (£)	Furniture Type	Description	Reporter Name	Option
1	Apartment	3	2018-02-23	11:15	950	Furnished	Spectacular views	Myles Findlay	Delete

Figure 10: User-defined data added to database table.

When the user clicks the 'Add to table' button the data entered by the user will be added to the database add will be shown in a table.
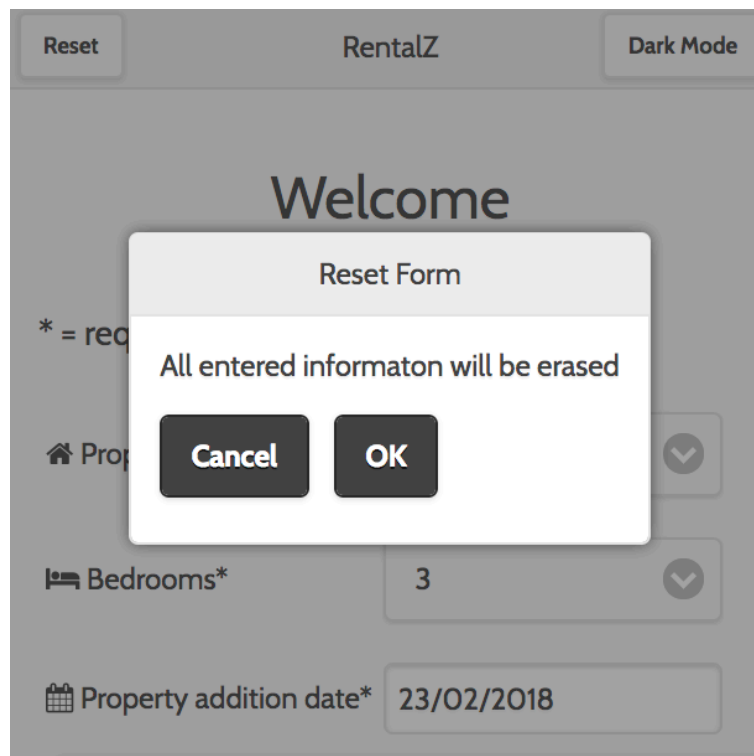
Figure 11: 'Reset Form' popup dialog

When the user clicks the 'Reset' button in the top bar, a popup dialog will ask them if they want delete all the information they had entered. If they tap the 'OK' button, all entered information will be removed and all select lists will be reset to their default values.
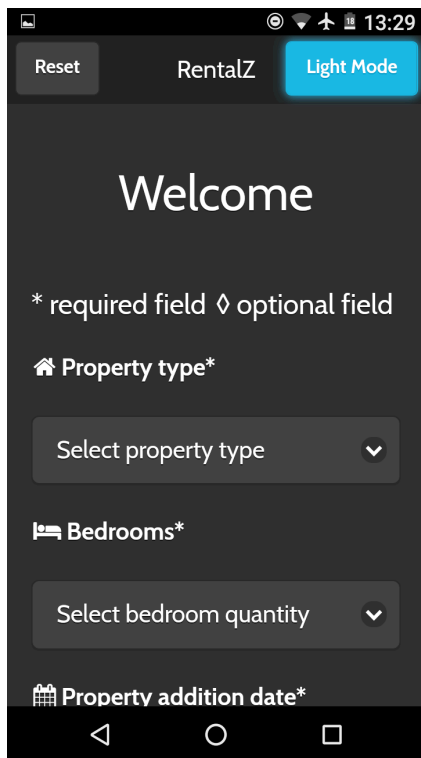
Figure 12: RentalZ App (dark mode - page 1 of 4)
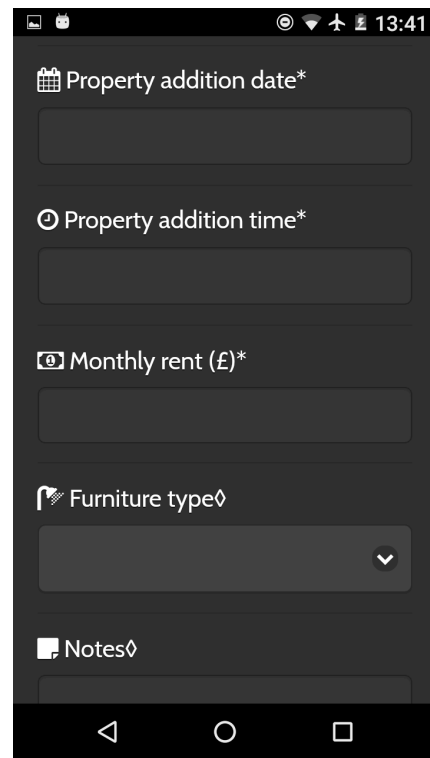
Figure 13: RentalZ App (dark mode - page 2 of 4)
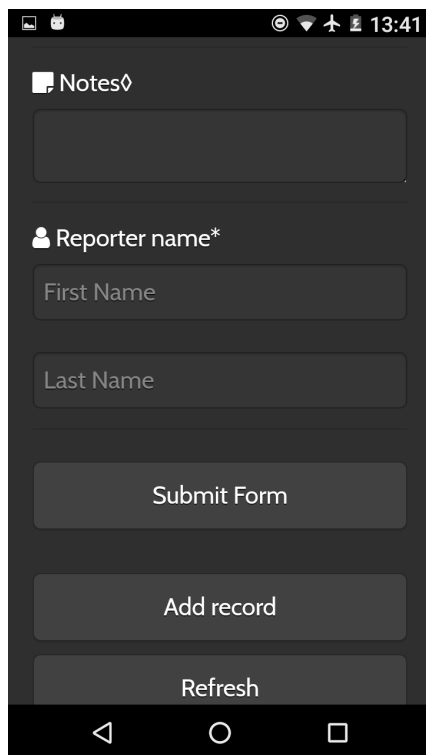
Figure 14: RentalZ App (dark mode - page 3 of 4)
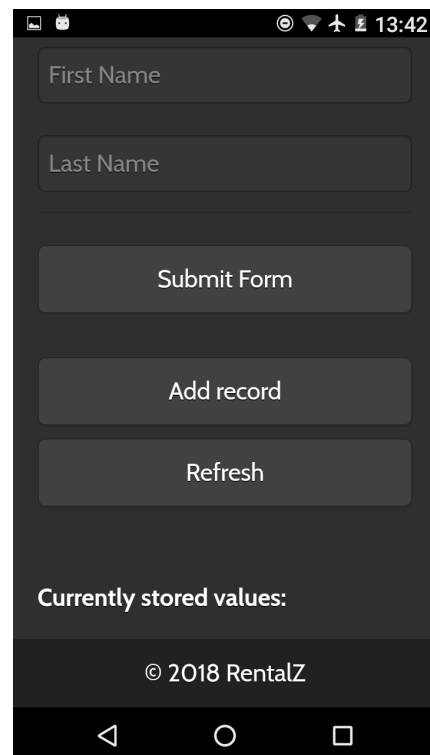
Figure 15: RentalZ App (dark mode - page 4 of 4)

Figure 16: RentalZ App (light mode - page 1 of 4)

Figure 17: RentalZ App (light mode - page 2 of 4)

Figure 18: RentalZ App (light mode - page 3 of 4)

Figure 19: RentalZ App (light mode - page 4 of 4)

2.2   Code that you wrote

File name: index.html

```
<!DOCTYPE html>
<html>

<head>
    <title>RentalZ</title>

    <!--Essential CDN (Content Delivery Network) links for JQuery Mobile
support-->
    <meta name="viewport" content="width=default-width; user-
scalable=no, initial-scale=1, maximum-scale=1, minimum-scale=1" />

    <!--Essential CDN (Content Delivery Network) links for JQuery Mobile
support-->
    <link rel="stylesheet"
href="http://code.jquery.com/mobile/1.4.2/jquery.mobile-1.4.2.min.css"
/>
    <script src="http://code.jquery.com/jquery-1.9.1.min.js"></script>
    <script src="http://code.jquery.com/mobile/1.4.2/jquery.mobile-
1.4.2.min.js"></script>


    <!--Cascading Style Sheets code for altering appearances of page
elements-->
    <link rel="stylesheet" href="styling.css">

    <!--JavaScript code for 'form reset' and 'theme toggling'-->
    <script type="text/javascript" src="programming.js"></script>

    <!--JavaScript code for Database and db-related functions-->
    <script type="text/javascript" src="databasing.js"></script>

    <!--
    <script type="text/javascript">
        app.initialize();

    </script>
-->

    <!--Font Awesome Icons support-->
    <link href="https://maxcdn.bootstrapcdn.com/font-
awesome/4.7.0/css/font-awesome.min.css" rel="stylesheet">
</head>

<body onload="init()">
    <div data-role="page" data-theme="a" id="pageform">
        <!-- header  -->
        <div data-role="header">
            <!-- When this button is clicked, show the
'resetformPopupDialog' PopupDialog -->
            <a href="#resetformPopupDialog" data-rel="popup" data-
position-to="window" data-transition="fade" class="ui-btn-left ui-
corner-all ui-shadow ui-btn-inline btnNormal">Reset</a>
```

```html
            <!-- Title of page -->
            <h1>RentalZ</h1>

            <!-- When this button is clicked, toggle between the light
and dark themes as necessary -->
            <a href="index.html" id="btn-theme" class="ui-btn-right
btnNormal" onclick="toggleTheme()">Dark Mode</a>
        </div>

        <!-- body/content/main content  -->
        <div data-role="content">
            <h1>Welcome</h1>

            <!--field types section-->
            <div class="ui-grid-a">
                <div class="ui-block-a">
                    <h3>&#42; = required field</h3>
                </div>
                <div class="ui-block-b">
                    <h3>&loz; = optional field</h3>
                </div>
            </div>


            <form>
                <table cellpadding="2" cellspacing="2">
                    <tr>
                        <td><i class="fa fa-home"></i> Property
type&#42;</td>
                        <td>
                            <select id="propertyType">
                                <option disabled selected
value="">Select property type</option>
                                <option
value="Apartment">Apartment</option>
                                <option
value="Bungalow">Bungalow</option>
                                <option value="Flat">Flat</option>
                                <option value="House">House</option>
                            </select>
                        </td>
                    </tr>
                    <tr>
                        <td><i class="fa fa-bed"></i> Bedrooms&#42;</td>
                        <td>
                            <select id="bedroomsCount">
                                <option disabled selected
value="">Select bedroom quantity</option>
                                <option
value="Studio">Studio</option>
                                <option value="1">1</option>
                                <option value="2">2</option>
                                <option value="3">3</option>
```

```
                                        <option value="4">4</option>
                                        <option value="5">5</option>
                                        <option value="6">6</option>
                                        <option value="7">7</option>
                                        <option
value="8&#43;">8&#43;</option>
                                </select>
                        </td>
                </tr>
                <tr>
                        <td><i class="fa fa-calendar"></i> Property
addition date&#42;</td>
                        <td><input type="date"
id="propertyAdditionDate"></td>
                </tr>
                <tr>
                        <td><i class="fa fa-clock-o"></i> Property
addition time&#42;</td>
                        <td><input type="time"
id="propertyAdditionTime"></td>
                </tr>
                <tr>
                        <td><i class="fa fa-money"></i> Monthly rent
(&pound;)&#42;</td>
                        <td><input type="number" min="100"
id="monthlyRent"></td>
                </tr>
                <tr>
                        <td><i class="fa fa-shower"></i> Furniture
type&#9674;</td>
                        <td>
                           <select id="furnitureType">
                                <option value=""></option>
                                <option
value="Furnished">Furnished</option>
                                <option
value="Unfurnished">Unfurnished</option>
                                <option value="Part furnished">Part
furnished</option>
                                </select>
                        </td>
                </tr>
                <tr>
                        <td><i class="fa fa-info-circle"></i>
Description&#9674;</td>
                        <td><input type="text" id="description"></td>
                </tr>
                <tr>
                        <td><i class="fa fa-user"></i> Reporter
name&#42;</td>
                        <td><input type="text" id="reporter"
placeholder="Name Surname"></td>
                </tr>
                <tr>
```

```
                                <td> </td>
                                <td>
                                    <!--button to add user-defined data into the
table-->
                                    <input type="button" value="Add to table"
class="btn-form" onclick="add()">
                                    <br>
                                    <!--<input type="button" value="Search"
class="btn-form" onclick="search()">-->
                                    <!--<br>-->
                                    <!--button to show all records data that
have been stored in the table-->
                                    <input type="button" value="Show all
records" onclick="displayAll()">
                                </td>
                            </tr>
                        </table>
                </form>
                <br>
                <br>

                <!--Table to show data entered-->
                <div id="result"></div>

                <!-- 'Reset form' PopupDialog-->
                <div data-role="popup" id="resetformPopupDialog" data-
overlay-theme="b" data-theme="a" data-dismissible="false">
                        <div data-role="header" data-theme="a">
                            <h1>Reset Form</h1>
                        </div>

                        <div data-role="main" class="ui-content">
                            <p>All entered informaton will be erased</p>
                            <a href="#" class="ui-btn ui-corner-all ui-shadow
ui-btn-inline ui-btn-b" data-rel="back" style="color: white;">Cancel</a>
                            <a href="#" onclick="resetForm()" class="ui-btn ui-
corner-all ui-shadow ui-btn-inline ui-btn-b" data-rel="back"
style="color: white;">OK</a>
                        </div>
                </div>
            </div>

            <!-- footer  -->
            <div data-role="footer">
                <h1>&copy; 2018 RentalZ</h1>
            </div>
        </div>
</body>

</html>
```

File name: styling.css

```
@import
url(http://fonts.googleapis.com/css?family=Cabin:400,300,500,700);
```

```css
h1 {
    /* align the text of the h1 header instance to the centre of the
screen */
    text-align: center;

    /* set the weight of the font of the h1 header instance to normal */
    font-weight: normal;
}

h2, h3, input, textarea, .ui-page, .ui-content, .ui-btn, td {
    /* the following applies to all 'h2', 'h3', 'input', 'textarea',
'.ui-page' and '.ui-btn' instances */

    /* set the font-family to the Cabin font */
    font-family: Cabin;

    /* set the font-weight to normal */
    font-weight: normal;
}

.ui-select span {
    /* align selected item of the selectmenu to the left */
    text-align: left;
}
```

File name: programming.js

```javascript
function toggleTheme() {
    var themes = {
            "a": "Light Mode",
            "b": "Dark Mode"
        },
        oldTheme = $(":mobile-page").page("option", "overlayTheme"),
        newTheme = oldTheme == "a" ? "b" : "a";

    // remove old class for 'ui-page' content, then apply the new class
to 'ui-page' content
    $("div[data-role='page']").each(function (index) {
        $(this).removeClass("ui-page-theme-" + oldTheme).addClass("ui-
page-theme-" + newTheme);
    });
    // remove old class for 'ui-bar', then apply the new class to 'ui-
bar'
    $(".ui-bar-" + oldTheme).each(function (index) {
        $(this).removeClass("ui-bar-" + oldTheme).addClass("ui-bar-" +
newTheme);
    });
    // remove old class for 'ui-body', then apply the new class to 'ui-
body'
    $(".ui-body-" + oldTheme).each(function (index) {
        $(this).removeClass("ui-body-" + oldTheme).addClass("ui-body-" +
newTheme);
    });
```

```javascript
        // remove old theme then apply the new theme to page content
        $(":mobile-page").page("option", "overlayTheme", newTheme);
        $(":mobile-page").page("option", "theme", newTheme);
        $(":mobile-page").page("option", "contentTheme", newTheme);

        // change 'btn-theme' button text as necessary
        $("#btn-theme").text(themes[oldTheme]);
}

function resetForm() {
        // perform all of the following if the 'OK' button of the 'Reset
Form' PopupDialog is clicked


        // set the selected option of the 'Property type' select list to
'Select property type' (disabled but default)
        var myselectA = $("select#select_propertytype");
        myselectA[0].selectedIndex = 0;
        myselectA.selectmenu("refresh");

        // set the selected option of the 'Bedrooms' select list to 'Select
bedroom quantity' (disabled but default)
        var myselectB = $("select#select_bedrooms");
        myselectB[0].selectedIndex = 0;
        myselectB.selectmenu("refresh");

        // clear all text from 'Property addition date' input field
        $('#input_propertyadditiondate').val('');

        // clear all text from 'Property addition time' input field
        $('#input_propertyadditiontime').val('');

        // clear all text from 'Monthly rent' input field
        $('#input_monthlyrent').val('');

        // set the selected option of the 'Furniture type' select list to
Furnished (default)
        var myselectC = $("select#select_furnituretype");
        myselectC[0].selectedIndex = 0;
        myselectC.selectmenu("refresh");

        // clear all text from 'Notes' textarea field
        $('#textarea_notes').val('');

        // clear all text from 'Report name' input field
        $('#input_reportername').val('');
}
```

File name: databasing.js

```javascript
var db = openDatabase('mytasks', '1.0', 'RentalZ Properties', 5 * 1024 * 1024);
// [database name, version number, text description, estimated database size in
Megabytes (MB)]

// function to initiate database
function init()
```

```
    "use strict"; //literal expression - tells the webpage that the JavaScript
code should be executed in "strict mode". With "strict mode", undeclared
variables (e.g.) cannot be used.

    db.transaction(function (tx) {
        tx.executeSql('DROP TABLE properties'); //delete table everytime the
page loads
        tx.executeSql('CREATE TABLE IF NOT EXISTS properties(id INTEGER PRIMARY
KEY AUTOINCREMENT, propertytype TEXT, bedrooms TEXT, propertyadditiondate TEXT,
propertyadditiontime TEXT, monthlyrent TEXT, furnituretype TEXT, description
TEXT, reporter TEXT)');
    });
}

// function to show all rows
function displayAll() {
    "use strict"; //literal expression - tells the webpage that the JavaScript
code should be executed in "strict mode". With "strict mode", undeclared
variables (e.g.) cannot be used.

    db.transaction(function (tx) {
        tx.executeSql('SELECT * FROM properties', [], function (tx, results) {
            var n = results.rows.length;
            var s = '<table cellpadding="2" cellspacing="2" border="1">';
//'table' opening tag
            s += '<tr><th>ID</th><th>Property
Type</th><th>Bedrooms</th><th>Property Addition Date</th><th>Property Addition
Time</th><th>Monthly Rent (&pound;)</th><th>Furniture
Type</th><th>Description</th><th>Reporter Name</th><th>Option</th></tr>';
//titles for table columns
            for (var i = 0; i < n; i++) { //for loop - when 'i' varibale is
less than n, increment by 1 + add new column with certain user-defined data
                var work = results.rows.item(i);
                s += '<tr>'; //'tr' (table row) (opening tag)
                s += '<td>' + work.id + '</td>'; //add 'id' value to this cell
                s += '<td>' + work.propertytype + '</td>'; //add user-defined
'propertytype' value to this cell
                s += '<td>' + work.bedrooms + '</td>'; //add user-defined
'bedrooms' value to this cell
                s += '<td>' + work.propertyadditiondate + '</td>'; //add user-
defined 'propertyadditiondate' value to this cell
                s += '<td>' + work.propertyadditiontime + '</td>'; //add user-
defined 'propertyadditiontime' value to this cell
                s += '<td>' + work.monthlyrent + '</td>'; //add user-defined
'monthlyrent' value to this cell
                s += '<td>' + work.furnituretype + '</td>'; //add user-defined
'furntiuretype' value to this cell
                s += '<td>' + work.description + '</td>'; //add user-defined
'description' value to this cell
                s += '<td>' + work.reporter + '</td>'; //add user-defined
'reporter' value to this cell
                s += '<td><a href="#" onclick="del(' + work.id +
')">Delete</a></td>';
                s += '</tr>'; //'tr' (table row) closing tag
            }
            s += '</table>'; //'table' closing tag
            document.getElementById('result').innerHTML = s; //store all table
content into the 'result' element
        });
```

```
    });
}

// function to add a new row, and insert data added/selected from form fields
as necessary
function add() {
    "use strict"; //literal expression - tells the webpage that the JavaScript
code should be executed in "strict mode". With "strict mode", undeclared
variables (e.g.) cannot be used.

    db.transaction(function (tx) {
        var propertytype = document.getElementById('propertyType').value;
//variable representing user-defined selection of the 'propertyType' element
(select list)
        var bedrooms = document.getElementById('bedroomsCount').value;
//variable representing user-defined selection of the 'bedroomsCount' element
(select list)
        var propertyadditiondate =
document.getElementById('propertyAdditionDate').value; //variable representing
user-defined selection of the 'propertyAdditionDate' element (date picker)
        var propertyadditiontime =
document.getElementById('propertyAdditionTime').value; //variable representing
user-defined selection of the 'propertyAdditionTime' element (time picker)
        var monthlyrent = document.getElementById('monthlyRent').value;
        var furnituretype = document.getElementById('furnitureType').value;
//variable representing user-defined selection of the 'furnitureType' element
(select list)
        var description = document.getElementById('description').value;
//variable representing user-defined selection of the 'description' element
(text area)
        var reporter = document.getElementById('reporter').value;
        tx.executeSql('INSERT INTO properties(propertytype, bedrooms,
propertyadditiondate, propertyadditiontime, monthlyrent, furnituretype,
description, reporter) values(?, ?, ?, ?, ?, ?, ?, ?)', [propertytype,
bedrooms, propertyadditiondate, propertyadditiontime, monthlyrent,
furnituretype, description, reporter], displayAll());
    });
}

// function to delete a row
function del(id) {
    "use strict"; //literal expression - tells the webpage that the JavaScript
code should be executed in "strict mode". With "strict mode", undeclared
variables (e.g.) cannot be used.

    db.transaction(function (tx) { //show dialog asking user if they want to
delete the row
        var result = confirm('Are you sure?'); //dialog messeage
        if (result === true) { //if 'OK' button is clicked, delete the
particular row
            tx.executeSql('DELETE FROM properties WHERE id = ?', [id],
displayAll());
        }
    });
}
```