**Report**

**1.** A description of each site file is as follow:

normalize.css and skeleton.css are part of the skeleton GUI framework, I added a few classes and types to skeleton.css, which are: ten-percent.column, five-percent.column, six-percent.column, eight-percent.column, header h5\_5 and decreasing the button padding from 30px to 18px.

add.php contains the HTML page and PHP code for the admin to add new records to the database.

The admin adds a new record using a POST form, values are sanitized, put into variables then added to the database through a MySQL query with an auto-incrementing ID.

The once POST is done the page reloads and if successful the admin will see the new record at the bottom of the table, if not then the appropriate message will be displayed.

Additionally the admin can edit a record by clicking its GET hyperlink or delete multiple by clicking their tick-boxes then the delete button on this page.

delete.php contains PHP code for deleting a record, it does not have its own GUI as it is called from other pages using “require ‘delete.php’;”.

If the entered ID is an integer and exists in the database then it will be deleted and the admin informed of that.

edit.php contains the HTML page and PHP code for the admin to edit an existing record, an ID is entered on the page or GET from another page, the admin edits the values, receives a message to indicate if successful or a message to indicate if not and why.

exists.php contains a PHP function to check if an ID exists in the database, the file is called using “require ‘exists’.php;” and the function is ID\_Exists($exists\_ID), this function is used in delete and edit operations on the home and edit pages where the admin inputs an ID.

I made this a separate file as multiple pages use it so it saves memory having it once instead of being on each page.

home.php contains PHP code and the HTML page for the home page, it links to the other pages and the admin can delete or edit a record from here if they know its ID.

search.php contain the PHP code and HTML page for searching the database using one of more values, alternatively the admin can enter no details to see all records.

From search the admin can edit a record by clicking its GET hyperlink or delete multiple by clicking their tick-boxes then the delete button.

**2.** I kept the database table simple as this assignment focuses on PHP, the database is one table with 8 columns.

I could have had more columns or linked tables to realistically represent a shopping database but I did not think it necessary.

**3.** Link to the application: http://student30397.bucomputing.uk/Assignment 1/console/

**4.** The self-assessment tick sheet is in the appendix of this document on page 5.

**APPENDIX**

**Sample database data and explanations.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID | Name | Description | Stock | Type | Style | Size | Price |
| 130 | Poka dot boxers | Boxers with a poka dot pattern on them. | 9 | Underwear | NA | S | 2.99 |
| 131 | Grey T shirt | Grey mens T shirt | 7 | Top | Summer | M | 5.99 |
| 132 | Blue Raincoat | A blue raincoat | 3 | Coat | Winter | L | 6.88 |
| 133 | Double stock | Double stock | 4.44 | Double stock | Double stock | Double stock | 1 |
| 134 |  |  |  |  |  |  |  |
| 135 | Partial |  | 3 |  | Partial |  |  |
| 150 | Decimal 3 price | Decimal 3 price | 6 | Decimal 3 price | Decimal 3 price | Decimal 3 price | 4.562 |
| 136 | Negative test | Negative test | -2 | Negative test | Negative test | Negative test | -6 |

Records 130 to 132 are template I have made and stored in a .sql file, they are used for testing the edit and delete operations and have been readded multiple times.

They would be accepted when inputted through the site as they do not break any of the data conditions/rules, Name, Description, Type, Style and Size are all valid strings, Stock is a valid integer and Price is a valid double.

Record 133 would not be accepted when inputted through the site as Stock only accepts integers, if entered the user would receive an error message.

Records 134 and 135 would not be accepted by the site as they have empty fields, the user would receive an error message to inform them that all fields must have a valid value if a field is empty.

Record 150 would be accepted but its price would be 4.56 instead of 4.562 as the PHP on the site rounds doubles to two decimal places.

Record 136 would not be accepted by the site as it has negative numbers, the user would receive an error message to inform them that numerical fields cannot be negative.

The data types used and why:

Name, Description, Type and Style are varchars as they need to hold strings that the user inputs, these strings may consists of numbers and special characters in addition to alphabetical characters.

ID and Stock are integers with ID being unique and autoincrementing, An autoincrementing number servers as a unique identifier/primary key for the database and for there cannot be two and a half of a product in stock so Stock is an integer.

Price is a double as currency tends to be measured to two decimal places and a double can go to decimal places unlike an integer.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Plan** | | | | | |
| Test | Purpose | Test data | Expected response | Actual response | Conclusion |
| Delete record from homepage | See if delete operation works from the home page | Go to home page, enter the ID of an existing record into the text box above the delete button then press the delete button. | A message will appear saying that the record with the entered ID has been deleted and the record will no longer be in the database. | A message appeared saying that the entered record was deleted and the entered record is no longer in the database. | Deleting a record from the home page works as intended. |
| Add a new record from the add page | See if add operation works on add page | Go to add page, enter data suitable for all fields (i.e. int for stock) then click the add button. | A message saying the record has been added will appear and the new record will be shown at the bottom of the list. | A message saying the record has been added appeared and the new record is shown at the bottom of the list. | Adding a record from the add page works as intended. |
| Entering record to edit from home page | See if the GET hyperlink on the home page to the edit page works. | On the home page enter an ID for an existing record in the text box above the edit button then click the edit button. | Will be taken to the edit page and the data associated with the record ID will be shown. | Was taken to the edit page and the data associated with the record ID is shown. | Inputting record ID to edit from the home page works as intended. |
| Search for all records | See if search function works when searching for all records. | Go to search page and click search button without entering data into any of the search fields. | All rows will be shown and ordered by ID. | All rows are shown and order by ID. | Searching for all records works as intended. |
| Search for record with a specific ID. | See if searching for specific ID works. | Go to search page and enter the ID of an existing record in the associated field then click the search button. | Only the record ID and the associated data will be shown in the table. | Only the record ID and the associated data is shown in the table. | Searching for a specific ID works as intended. |
| Edit and delete ID that does not exist | See if the ID\_Exists function works as intended. | Go to home page, enter an ID number that does not exists in that database into the delete field then press the delete button, do the same for the edit field and the edit field on the edit page. | A message saying that a record with the entered ID does not exist will appear each time. | A message saying that a record with the entered ID does not exist appeared each time. | The ID\_Exists function works as intended. |
| Try to add a new record from the add page but with no name. | See if code to detect that no value has been entered and display appropriate message works as intended. | Go to add page and enter valid data into all of the add record fields except name then press the add button. | A message will appear saying that there must be a value in all fields and an incomplete record was not added. | A message appeared saying that there must be a value in all fields and an incomplete record was not added. | The no value detection code works as intended when adding a value. |
| Delete multiple records from search page | See if the batch delete code works as intended. | Go to search page, click search button to show all records, tick the checkboxes for 3 records then click the delete button. | A message saying that the selected records (showing their IDs) have been deleted will appear and those records will no longer be in the database. | A message saying that the selected records (showing their IDs) have been deleted appeared and those records are no longer in the database. | Batch delete code works as intended. |
| Get to edit page for a record from search page. | See if GET hyperlink on search page to edit page works as intended. | Go to search page, click search button to show all records, click the edit hyperlink next to a record. | Will be taken to edit page for the selected record and its values will be shown. | Was taken to edit page for the selected record and its values were shown. | Search page GET hyperlink to edit page works as intended. |
| Adding record with invalid values. | See if invalid value detection code on add page works as intended. | Go to add page and put letters in the price and stock fields with the other field having valid values. | Record will not be added and a message will appear saying that all fields must have a valid value. | Record is not added and a message appeared saying that all fields must have a valid value. | Invalid value detection code works as intended. |

**Entity Relationship Diagram**

|  |
| --- |
| Items |
| P\* Item\_ID  Item\_Name  Item\_Description  Item\_Type  Item\_Style  Item\_Size  Item\_Stock  Item\_Price |
| PK\_Item\_ID |

**WP Assignment 1 Self-Reflection Sheet**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Yes (completed)** | **Partially completed** | **No (not submitted)** |
| **Report** |  | | |
| Database design | X |  |  |
| Test cases | X |  |  |
| Sample data | X |  |  |
| **System** |  | | |
| Search function | X |  |  |
| Edit function | X |  |  |
| Add function | X |  |  |
| Delete function | X |  |  |
| Additional features | X |  |  |
| Robustness & Security | X |  |  |

**In your own opinion, what is the best part of your work?**

|  |
| --- |
| The function for checking if an ID exists in the data base when an integer is inputted. |

**In your own opinion, what is the worst part of your work?**

|  |
| --- |
| There may be too many conditions and ifs on a few of the pages. |

**In your own opinion, what is the part you struggled most with?**

|  |
| --- |
| Syntactic errors resulting in time lost to troubleshooting when the issue was a missing bracket or a quotation mark in double quotes that Eclipse and Netbeans did not detect. |