5COSC024W Server-side Web Development Mock ICT01 – Model Answers

Question 1

Referring to the code on line 16 and line 18.

Explain what `previd` is (line 16), how it gets created and what it contains.

[4 Marks]

- previd is a URL parser or query string.
- It is appended to a URL on the previous page inside an anchor tag as the id of the selected product is passed by URL using the GET method.
- On the previous page (which is not available here) there would have been a <a> tag like this:
  `Echo "<a href= nicepage.php?previd=".arrayp['prodId']. ">`
- Therefore, the id number would have been passed through the URL using the URL parser (or query string) previd using the GET method.

Question 2

Referring to the code on line 16 and line 18.

Describe what happens to `previd` and `$id` if the user the selects the 'Cookie Jar' in the ***pro*** table presented on the "Sample Data and PHP Code" document.

[4 Marks]

- When the user clicks on the cookie jar on the previous page, the value 6211 would be appended to the URL as a URL parser (or query string) previd.
- When the user hovers the anchor, they can see the id being dynamically captured.
- When the user clicks on the anchor, they can see the id being attached to the URL when accessing nicepage.php.
- The value 6211 would then be captured with the $_GET super global variable.
- The value 6211 would then be assigned to the local variable $id.
- The product id 6211 is then retrieved from the table using the query so that all the details of the cookie jar can be displayed on this page nicepage.php

Question 3

Referring to the code from line 18 to line 20.

Explain what `$SQL` is (line 18), how it gets created and what value it contains.

[4 Marks]

- $SQL is a local variable of type string.
- It gets assigned a SQL query.
- It contains the SQL query that retrieves all the details of the product for the product whose id matches the id passed by URL from the previous page.
- Therefore, the SQL query retrieves the products selected by the user on the previous page.

Question 4

Referring to the code from line 18 to line 20.

Explain what `$runSQL` is (line 19), how it gets created and what value it contains.

[4 Marks]

- $runSQL is a local variable that contains the outcome of the execution of the SQL query.
- It gets created using the mysqli_query function that executes SQL queries for a specific connection or the mysql_error function if the execution of the SQL query is not successful.

Question 5

Referring to the code from line 18 to line 20.

Explain what `mysqli_fetch_array` is (line 20) and how it is used here.

[4 Marks]

- mysql_fetch_array is a PHP function that fetches the database records retrieved by the execution of the SQL query and stores them in an array called $details.
- In this case it retrieves the details of the products selected by the user on the previous page and store them on the array $details.
- $details is subsequently used to:
    - Display the product id, the name, the description, the price and the stock level.
    - Display an image based on the name of the file for this product stored in the database table.

Question 6

Referring to the code from line 20 to line 27.

Explain what `$details` is (it appears 7 times on line 20 to line 27), how it gets created and what value it contains.

[4 Marks]

- $details is an array of records.
- It was created by fetching the results of the execution of a SQL query that retrieved details of a specific product.
- To retrieve specific items, you use the name of the array and in square brackets the name of the column of the database table.
- On line 22 it is used to display the product id as plain text as a catalogue number.
- On line 23 it is used to display the product name as plain text in capital letters.
- On line 24 it is used to display the product description as plain text.
- On line 25 it is used to display the product image as located in the “images” directory.
- On line 26 it is used to display the product price with a £ sign.
- On line 27 it is used to display the product quantity as available in stock

Question 7

Referring to the code from line 20 to line 27.

Explain what `strtoupper` is (line 23) and how it is used here.

[4 Marks]

- strtoupper is a PHP function that converts a string in capital letters.
- Here it is used to display the product name retrieved from the database table and currently in the array $details in capital letters.

Question 8

Referring to the code from line 20 to line 27.

Explain in detail line 25.

[4 Marks]

- Line 25 displays an HTML image with a dynamic pathway.
- The src attribute of the image tag specifies the pathway for the images to be located.
- In this case, it locates the “images” folder and inside this folder the name of the image as retrieved from the database and currently in the array $details.

Question 9

Referring to the code from line 20 to line 27.

On line 22, could `$id` be used instead of `$details['prId']` to display the catalogue item number? Explain.

[4 Marks]

- Yes, they can be used interchangeably here.
- $id contains the id of the product as selected by the user, appended to the URL and passed thnrough using the GET method.
- $details['prId'] contains the id of the product retrieved from the database after the query has pulled out that specific product selected by the user.
- So they are definitely the same thing.

Question 10

Referring to the code from line 29 to line 34.

Explain what `method` is (line 29) and how it is used here.

[4 Marks]

- Method is an attribute of a <form> tag that specifies the way the data captured in the form is going to be sent to the location specified in the action attribute.
- If the method is GET then the value passed through will be visible in the URL and the $_GET superglobal variable will be used to capture the value.
- If the method is POST then the value passed through will NOT be visible to the user and the $_POST superglobal variable will be used to capture the value.

Question 11

Referring to the code from line 29 to line 34.

Explain what `action` is (line 29) and how it is used here.

[4 Marks]

- Action is an attribute of a <form> tag that specifies the pathway of the PHP file to be called upon when the user submits the data in the form.
- It defines the location of the PHP file to be called from the server when the user clicks on the submit button i.e. the file where the user is going to be directed and where the data captured in the form is going to be processed.

Question 12

Referring to the code from line 29 to line 34.

Referring to line 33, explain why `hidden` needs to be used here and why `value` is set to `$id`.

[4 Marks]

- The form created here captures the number of items the user wants to place an order for.
- Therefore, it essentially passes one value through the form using the POST method: the number of products the user wants.
- However, an additional value needs to be passed through: the id of the product. This is so that the specific product actually selected by the user can be identified and its details later retrieved.
- This is what a hidden field is created: this is like a form element, but it is invisible to the user.
- The value it carries through is the $id, which contains the id of the selected product as retrieved from the previous page using the GET method.

Question 13

Referring to the code on line 58 and line 59.

Explain what `newid` is (line 58), how it gets created and what it contains.

[4 Marks]

- newid refers to the hidden field that was created on line 33 as part of the form.
- It passes through the id number of the selected product using the post method.
- It is hidden as it is the second data item passed on top of the number of products the user selects.
- It is captured on line 58 using the $_POST super global variable.

Question 14

Referring to the code from line 58 to line 61.

Explain what `$theid` is (line 58 and then line 61), how it gets created and what it contains.

[4 Marks]

- $theid is a local variable
- It gets assigned the product id passed by form (POST method) through the hidden field newid created as part of the HTML form on the previous page.
- It is assigned the id as captured using the superglobal variable $_POST

Question 15

Referring to the code on line 58 and line 59.

Explain what `newqu` is (line 59), how it gets created and what it contains.

[4 Marks]

- newqu refers to the text field that was created on line 31 as part of the HTML form.
- It passes through the number of items required by the user for the selected product using the post method.
- It is captured on line 59 using the $_POST super global variable.

Question 16

Referring to the code from line 61 to line 92.

Explain in what situation the condition on line 61 can be verified and thus trigger the code on line 63.

[4 Marks]

- The condition on line 61 is verified if the super-global variable that captures the id of the selected product passed through the hidden field is not set and therefore does not contain a value.
- If that is the case, it means that the user is not trying to add a new product into the basket but that they are merely accessing the basket just to view it (without trying to update it).
- Therefore line 63 displays a message that says that nothing new is added and to only show the existing content of the basket.

Question 17

Referring to the code from line 61 to line 92.

Explain in what situations the condition on line 67 can be verified and thus trigger the code on lines 69, 70 and 71.

[4 Marks]

- The condition on line 61 is verified if the local variable that captures the quantity of items product passed through the text field in the HTML form does not contain a value or contains a 0 value.
- If that is the case, it means that the user has either not entered anything in the text box or entered the value 0 and click the submit button. (Of course, no other types of characters are checked here)
- Therefore line 69 and 70 will display an error message and a link back to the product page.

Question 18

Referring to the code from line 61 to line 92.

Explain what happens if the `else` on line 73 is verified and thus if the code directly following it is triggered.

[4 Marks]

If the else on line 73 is verified, then:

- the id of the selected product does exist (therefore the user is definitely adding a new product onto the basket)
- the number of items has been entered and it is not 0.
- Providing that this number of items is actually a numerical value then we can proceed with line 80 that compares it to the quantity available in stock.

Question 19

Referring to the code from line 75 to line 90.

Why does a SQL query need to be written on line 75?

[4 Marks]

- The SQL query is written to retrieve the quantity of products stored in the database table and therefore available in stock.
- This number of items available in stock is needed to ensure that the number of products entered by the user is available.
- This stock value is used to proceed with the comparison on line 80.

Question 20

Referring to the code from line 75 to line 90.

Explain what `$info` is (line 78), how it gets created and what it contains.

[4 Marks]

- $info is a local variable of type array.
- It is an array of records that only contain one record i.e. the number of items available in stock for this product.
- By writing $info['prQuantity'] we can specifically refer to the stock level for this product.

Question 21

Referring to the code from line 75 to line 90.

Compare and contrast `$thequ` and `$ourqu`. Explain what they are, how they get created and what they contain.

[4 Marks]

- $thequ is a local variable of type integer that contains the number of items the user wants.
- It was created using the $_POST superglobal variable to capture the value passed by form in the text box.
- $ourqu a local variable of type integer that contains the number of items that is available in stock.
- It was created using the array of records $info that feteched the results of the execution of the SQL query to retrieve the quantity available in the table.
- On line 80 we compare them:
    - If the number of items the user wants is greater than the number of items in stock, an error message is displayed.
    - If the number of items the user wants is less than the number of items in stock, then the product id and the number of items entered by the user are stored in the session array.

Question 22

Referring to the code on line 42 and line 89.

Explain what `$_SESSION['storage']` is, how it gets created and what is it used for.

[4 Marks]

- $_SESSION['storage'] is a session array i.e. an associative array that consists of a key (i.e. the index) and a value (i.e. the content).
- For every iteration of the user in the web site when they are selecting and adding products into the basket it will store the id of the selected product as a key and the number of items as a value.
- It is a construct that is accessible from multiple pages providing that they include session_start()
- It stores information in memory for the entire duration of the session.
- It can be used to store data relevant to the product required by the user like in this case where it is going to store the id number and the number of items the user wants for all required products. These will then be used to retrieve the product details to build the basket.
- It can be used to store user information like the user id, their name and user type.

Question 23.

Referring to the code on line 42 and line 89.

Explain how `$theid` and `$thequ` are used to populate the `$_SESSION['storage']` structure. Use examples taken from the ***pro*** table on the "Sample Data and PHP Code" document to illustrate your answer.

[4 Marks]

- Line 89 creates a new cell in the session array for the latest product to be added.
- $the id contains the id number of the latest required product to be stored as a key (i.e. an index).
- $thequ contains the number of items for the latest required product to be stored as a value (i.e. the content).
- If the user is trying to purchase 2 items of the "silver juicer" product, a new cell is created in the $_SESSION['storage'] session array.
    - This new cell will be indexed with the value of the id of the product (value 5856 for the silver juicer).
    - The cell will actually store the value of the number of items the user wants for this product (2 in this example).

Question 24.

General Question

Explain how HTML, CSS, PHP and SQL are combined together to create dynamic server-side web applications.

[4 Marks]

- HTML is used to display web documents in a Web browser. It provides a way to create structured documents by defining elements that are then used as building blocks for the page: headings, paragraphs, lists, links, images, from elements, etc. It is used to create static pages whose content only change when the web administrator decides to change it.
- CSS is used for defining the presentation of a web document authored in a markup language like HTML by formatting elements such as layout, colours, and fonts.
- PHP (Hypertext Preprocessor) is a server-side scripting language that is embedded within HTML and allows for the development of dynamic web applications that supports web-database integration. It is used to create dynamic pages whose content change depending on the user's behaviour.
- SQL is embedded within PHP to enable web-database integration as follows
    - Create a $SQL variable and assign a SQL query to it.
    - Use the mysqli_query function to run the query.
    - Use the mysqli_fetch_array function to extract the data and store it in an array of records.
    - Use the array to render dynamic elements (PHP) within a static shell (HTML) e.g. image, anchors, plain text, etc.

- Using SQL within PHP supports the interactions between a front-end web app and a back-end database by implementing the CRUD operations
    - CREATE: write an INSERT INTO query in SQL to add new records into a database table.
    - READ: write SELECT query in SQL to retrieve existing records from a database table.
    - UPDATE: write an UPDATE query in SQL to edit existing records located in a database table.
    - DELETE: write a DELETE query in SQL to remove EXISTING records from a database table

Question 25.

General Question

Explain how CSS is used to create a consistent look-and-feel across a web app.

[4 Marks]

- CSS enables the separation of presentation and content.
- It allows the centralisation of presentation elements (i.e. definition of the formatting of the layout of a web site) in a separate file.
- It enables the sharing of formatting by multiple changes and edits to be made for the entire web application.
- Therefore, it allows centralisation, reusability, and updatability to support the consistency of the look-and-feel of the application.