CS 120 Web Programming
Project 3

Your assignment is to create an interactive e-commerce website utilizing your skills in HTML/CSS/JavaScript/MySQL/PHP. You should not build the entire site – only the pages identified below. The list of pages required is listed below.

Note: the total number of points intentionally exceeds 100. If you complete everything, you will have extra credit.

Tasks

Your task is to create four pages that would be part of an e-commerce website as well as a database to support the website. Each page must have a logo and navigation to each of the pages. You are not required to build a home page.

Database: (25 points)

Use MySQL as the database with PHP.

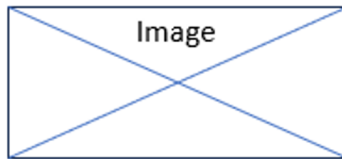*Deliverable for this part is a screen shot of the database tables.*

Create a database that can store products and orders. You may choose any product type (food, cosmetics, books, clothing, etc). You must have at least 8 products.

- Product data should include: id, name, price, description, url of a product image

- Each product image should be able to be rendered at the same size (ie, they need to be the same aspect ratio). Product images greater than 70k in size should have their quality or dimensions adjust using https://birme.net or other photo editor.

- The images should be uploaded to the hosting server in a subfolder of the project root folder. Name the subfolder: "images".

- Orders should include: id, item id, order date, quantity for up to 5 items.
  Optionally, add an Assignment table to allow unlimited items to be ordered.

Page 1: Products page (30 points)

Display all products. Create a "card" for each product to include the image, name of product, price, and two buttons labeled: "Add to Cart" and "More".

A *possible* layout for each product is as follows:

Widget $10.00

[Add to Cart]   [More]

not very convinient

Add to Cart should add the product to a shopping cart list (use a session variable or local storage-do not store in the database) and proceed to the cart page. If an item is added to the cart more than once, add one to the quantity already in the cart. ✓

"More" should reveal a div positioned near or below the buttons containing a description of the product and the text on the button face should change to "Less".
Clicking "Less" should hide the description and change the text on the button back to "More". ✓

Page 2, 3: Cart page and Thank you page (40 points) ✓

The cart page should display a list of all items currently in the cart. For each item, display the name, quantity, price and total cost for that item. Do not show the image.

Each item should also have a button labeled "Remove from cart" adjacent to it that will delete that item.

Below the list of cart items, show the total for the order (no shipping or tax is needed). There should also be two buttons: "Check Out" and "Continue Shopping".

- When "Check Out" is clicked, go to a **thank you page** which will store the order in the database and clear the cart. Include a message that thanks the user for their order. List the order total and expected ship date (calculate and display the date to be 2 days from now)

- "Continue Shopping" will go back to the Product page without clearing the shopping cart.

Page 4: Orders Page (15 points) ✓

Show all orders including the date, order id, order total. For each item in the order, display: items ordered/quantity/total cost for that item.

The most recent order should be displayed first. ✓

Suggestion: Populate a few orders in the database to test this page PRIOR to creating the product and cart pages.

Deliverable Worksheet

✓ All code files (combine into a compressed/zip file – do not add this deliverable worksheet to the zip file)

✓ Paste screenshots of each table in the database below

URL for product page: ___ https://yuqingw.sgedu.site/Project3/products.php ___

URL for orders page: ___ https://yuqingw.sgedu.site/Project3/orders.php ___ ? ✓≈ ···

Project requirements- check off those you completed.

___ ✓ Database created to store orders and products

___ ✓ Products page

___ ✓ Cart page

___ ✓ "Thank you" page

___ ✓ Orders page

Are there any extra features you added that are not listed above?

I added a timestamp parameter (?v=time()) on the Orders page link to prevent browser caching and ensure the page always loads fresh data.

What was the most interesting part of this project?

The most interesting part was implementing the logic that writes orders and order items into the database. Seeing the cart instantly turn into a structured order record made the project feel like a real e–commerce workflow.

Product table has 8 data in it

| | | | | Browse | Structure | SQL | Search | Insert | Export | Import | Operations | Triggers |

✔ Showing rows 0 - 7 (8 total, Query took 0.0003 seconds.)

SELECT * FROM `product`

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

				id	name	price	description	image_url
☐	✏ Edit	Copy	⊘ Delete	1	iPhone 16	899	Latest smartphone with A18 chip and dual-camera sy...	iPhone.jpg
☐	✏ Edit	Copy	⊘ Delete	2	iPad Pro 11"	999	M4 chip, 120Hz display, supports Apple Pencil	iPad.jpg
☐	✏ Edit	Copy	⊘ Delete	3	MacBook Air 13"	1099	Lightweight laptop with M2 chip and long battery l...	MacBookAir.jpg
☐	✏ Edit	Copy	⊘ Delete	4	MacBook Pro 14"	1999	High-performance laptop with M3 Pro chip	MacBookPro.jpg
☐	✏ Edit	Copy	⊘ Delete	5	Apple Watch Series 10	399	Advanced health and fitness tracking	AppleWatch.jpg
☐	✏ Edit	Copy	⊘ Delete	6	AirPods Pro (2nd Gen)	249	Active noise cancellation and spatial audio	AirPods.jpg
☐	✏ Edit	Copy	⊘ Delete	7	Apple Pencil Pro	129	Precision stylus for drawing and note-taking	ApplePencil.jpg
☐	✏ Edit	Copy	⊘ Delete	8	iMac 24-inch	1299	24-inch 4.5K Retina display all-in-one desktop, Ap...	iMac.jpg

Order table

| | Browse | Structure | SQL | Search | Insert | Export | Import | Operations | Triggers |

Table structure | Relation view

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action		
☐	1	**order_id** 🔑	int			No	*None*		AUTO_INCREMENT	✏ Change	⊘ Drop	More
☐	2	**order_date**	datetime			No	*None*			✏ Change	⊘ Drop	More
☐	3	**total_price**	decimal(10,0)			No	*None*			✏ Change	⊘ Drop	More

Order_item table

| | Browse | Structure | SQL | Search | Insert | Export | Import | Operations | |

Table structure | Relation view

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action		
☐	1	**item_id** 🔑	int			No	*None*		AUTO_INCREMENT	✏ Change	⊘ Drop	More
☐	2	**order_id**	int			No	*None*			✏ Change	⊘ Drop	More
☐	3	**product_id**	int			No	*None*			✏ Change	⊘ Drop	More
☐	4	**quantity**	int			No	*None*			✏ Change	⊘ Drop	More