Instructions

- o Run the pet-shop.sql file in your PHPMyAdmin to create the "pet_shop" database, which has the "grooming" table where you will store your grooming appointments.
    - o When connecting to the db, be sure to use "root" as the username, "pwdpwd" as the password, and "pet_shop" as the database. Your php code should therefore look something like this: '"$db = new mysqli('localhost','root','pwdpwd','pet_shop');"'
- o Your homepage file name should be called "index.php"'
- o The site should have a header, body and footer. Use the HTML5 header and footer elements. Be sure to use PHP includes to include your header and footer throughout your pages.
- o In the header, include a navigation bar to the main pages:
    - o About Us
    - o Store Location
    - o Grooming
- o In the header, also include a link to "Contact Us". This link opens up a page with a contact form. The contact form should be submitted using Ajax. The fields should be validated prior to submission to the PHP script with JavaScript. The PHP script will create a new email that is sent to the shop owner (yourself for now). The contact us form should have the following input fields:
    - o First Name
    - o Last Name
    - o Email Address
    - o Message
- o The footer should include hours of operation and link to other local services that Sandy's customer's would be interested in, such as veterinaries and kennels.
- o This site will also include a simple backend (or admin) interface to view grooming form submissions, such that you can manage the submissions by clients. On the admin page (located at /admin), you will list each grooming request, with a link to edit and delete existing grooming requests. You will need to create the necessary HTML and PHP code to handle each form submission:
    - o Create new grooming service requests
    - o Edit/Update submitted grooming service requests
    - o Delete submitted grooming service requests if they have been cancelled.
- o All links to other pages on the site, to images, and to stylesheets should use relative paths.
- o The site can optionally use jQuery for performing the Ajax operations.
- o The site should fit cleanly for all devices:desktop browser, tablet, smartphone.

The main pages should include some content, such as:

- o The About Us page should include a list of the staff that work as Sandy's Pet Shop.
- o The Store Location page should include directions to the store and the hours of operation. You might also include a Google map with an indicator to the pet shop.
- o The Grooming page will include information on the pet grooming services and will have a form for Sandy's customers to sign up for a pet grooming appointment. The data for the grooming form is stored in the MySQL database table named "grooming" (the database itself will be named "pet_shop" when you run the SQL script included in the class files to create the database). Be sure to validate the form submission with JavaScript. The form should include the following fields:
    - o First Name (required)
    - o Last Name (required)
    - o Address (required)
    - o City (required)
    - o State (required)
    - o Zip code (required)
    - o Phone Number (required)
    - o Email Address (optional)
    - o Type of Pet. If dog is selected, a second select list should show up with a list of Breeds. (required)

- o   Pet's Name (required)
- o   Checkbox for neutered/spayed (optional)
- o   Pet's age (optional)

Setup

To set up your project:

- o   Use the existing setup from your [Introduction to PHP course](#)
- o   [Download the project files](#)
- o   Extract the project files into a new folder in your web root. This folder should be called "pet-shop".
- o   Open up your browser to the pet shop home page: http://localhost/pet-shop
- o   Open up your browser to the pet show grooming administration page: http://localhost/pet-shop/admin
- o   When you open your browser up to these pages, you will initially get an error indicating the page does not exist. You will need to write the HTML necessary for both the main site for Sandy's pet shop, as well as the grooming administration pages.

Testing Your Project

To test your project:

- o   View the source in your browser and review the code there. This shows you the HTML that your PHP code generated.
    - o   Make sure your code starts with a DOCTYPE declaration, preferably `<!doctype html>` . There should only be one DOCTYPE declaration per page.
    - o   Validate the code by copying it and pasting it in [[https://validator.w3.org/#validate_by_input|the W3C Validator]]. If you get any errors, review them and try to fix them before submitting the project.
- o   Do you have any broken links?
- o   Do all your internal links, stylesheets, and images `src` attributes use relative paths?
- o   Do all your pages have titles?
- o   Are you using CSS (and not HTML) for formatting. For example,
    - o   Don't use <br> tags or <p> tags to create vertical space.
    - o   Don't use deprecated tags (e.g., `<center>`, `<frame>`, `<font>`, `<u>`, `<b>`) and attributes (e.g., `align`, `background`, `bgcolor`, `border`, `hspace`, `vspace`, `width`)
- o   Does your design work well in different dimensions? Resize your browser window to different sizes to make sure; ideally, check your work on a smartphone or mobile emulator.
- o   Forms
    - o   Go to your contact page and submit the contact form without filling anything in. It should give you an error.
    - o   Test your other validation code in the contact form as well.
    - o   Fill out the contact form with valid data and submit it. Does it submit using Ajax? Does an email get sent?
    - o   Test your Grooming form in the same way.
- o   Do you have header and footer include files? Do they include the correct content?
- o   Did you carefully follow the instructions for the '''Grooming''' page in the '''Requirements''' (above)?
- o   Do you have an '''Admin''' interface at /admin and does it work as per the instructions in the '''Requirements''' (above)?

Submitting Your Project

After you have completed all of the requirements of the project:

- o   Create an archive (zip file) of your entire project "topic" folder. You can use WinZip, or whatever compression tool you prefer to create the .zip file.
- o   Email the file to project@webucator.com.
- o   Please allow 10 business days for us to review the exam submission.

- You must complete the project before the expiration date of your course (12 months after you started). We estimate it will take about 40 hours. Be sure to leave yourself enough time.
- **IMPORTANT:** Please be sure to complete ALL the requirements of the project before submitting it. If for any reason you are not able to complete one or more of the requirements and wish to submit without that portion complete, please explain that in the email when you submit the project. If a project is incomplete and there is no explanation as to why, it can not be graded.

Grading

Your grade will be calculated based on the following categories, with the amount that each category contributes to your overall grade:

HTML: 30%

- Valid: 5%
- Links work and use relative paths when possible: 5%
- Forms are structured correctly: 5%
- No deprecated tags and attributes: 5%
- Structured intelligently: 5%
- All required content included: 5%

CSS: 25%

- Used for layout as well as formatting: 10%
- Valid: 5%
- Design works in different browser dimensions and device widths: 10%

JavaScript: 20%

- Form validation on Contact Form: 5%
- Contact Form submitted using Ajax: 5%
- Form validation on Grooming Appointment: 10%

PHP and SQL: 25%

- Header and Footer includes used appropriately 5%
- Contact Form submission works: 5%
- Email generated by contact form: 5%
- Grooming form submission works: 5%
- Admin page works works appropriately: 5%

Rules

This project is meant for you to use your own skills and knowledge. This means that we expect the work to be your own work. We also expect that you will want to look some stuff up. Please feel free to use your course textbook and the course content to help you along. You may also use the Internet as a source of help, especially for looking up documentation and errors.

Note that the instructor is not a resource during this project. The purpose of the project is to evaluate how well you can do without access to the instructor.