

WWW110 PHP Homework 1

Creature Generator

Due: December 14 @ 8am it must be available on your Github!

Worth: 50% of your grade for this unit (WWW110)

You are creating a 2 page “creature generator” based on the files provided. The first page is already constructed for you (step_one.php) however you will need to modify and duplicate it to successfully complete the assignment.

You are allowed *and encouraged* to alter any of the HTML/CSS as you see fit, including adding id's or classes to the code. There will be a 2% deduction for every error the w3c validator reports, on every page. If you want to leave an error in there, you can, but you must leave a code comment beside the offending line explaining why. Making informed choices that sometimes break the rules is just part of being a webmaster.

The creature generator is a 2 step process. On page one (step_one.php) you will be asking the user to provide their name, and if they will be turning into an alien, or a robot. The second page should say:

“Thanks [name]Eric[/name].

Today is [dayname] [monthname] [daynumber][daysuffix], [year] and it's been a busy day. But don't worry!

[message]Our scientists are working overtime gathering spare claws and feathers to build your alien.[/message]

When you are ready to see the results, click here”

Where [name] is the name they provided on the first page, and [message] IS YOUR OWN CLEVER MESSAGE. *Seriously*, don't type what I provided, write up something witty, funny, or at least explanatory. If you don't include some type of witty message, no matter how lame, you will lose 5%. Part of being a webmaster is engaging your audience, and this is part of the job.

If a user leaves their name blank, or doesn't select an alien/robot choice, they should be provided an error message and will not be able to proceed. This logic should be done via PHP (although Javascript form enhancements are absolutely welcome). “click here” can be a regular hyperlink or a button. Clicking it should reveal a hidden portion of the page (you can use Javascript to do this, ie: jQuery slideDown()).

Specifically:

- You should make up a robot/creature name based on the name they provided. A simple “Name-bot”, or “Namezula” is fine. The name **MUST** be in the format capital first letter, lowercase everything else, regardless of how the person entered it. PHP string manipulation will be needed.
- There should be a short description about the creature (see below)
- There should be a picture of the creature (robot.png and alien.png have been provided, but it'd be ‘cooler’ if you used your own properly sized and optimized images).

This description loaded up is special. You need to create a pool of at least 4 descriptions for robots, and 4 for aliens. If your user selected alien, one of the 4 is randomly shown on the screen. You will need to look up how to randomly do this - it was explicitly not shown in class. There are several possible approaches, try to evaluate a few possibilities and choose

the best one for you. If you try multiple things, please comment out the ones you aren't using to show that you tried them. Robot descriptions should not show if the user selected alien in step one, and vice versa. Hint: a structure called an "array" is probably the most straight-forward way to begin this, but is not required.

The page should also automatically email a message formatted as so to eric.chen@senecacollege.com when it is visited. You can use PHPMailer or PHP's native mail function.

Name: [name]!
Creature Type: [creature]
[creature description that displayed on screen]

Each page should share common elements included through one of the 4 include functions of PHP (choose one that you feel is appropriate).

"EXTRAS"

1. It'd be neat if the form on step 1 included Javascript validation.
2. The name field should be explicitly sanitized to prevent HTML tags
3. It'd be neat if the images were random also, or tied to the descriptions.
4. Indicate which coding style you are following in a comment at the very top of every file.
5. The look and feel of the application is very basic. It'd be nice if it was enhanced, or even Bootstrap powered.

The marks break down as follows:

- common element inclusion 5%
- harvesting name from step one into step two including sanitation 20%
- creature name 15%
- correctly loading a random description for the creature 30%
- correctly coded mailing procedure 15%
- following good coding procedures (this means bracket indention, properly naming variables, being consistent in spacing and placement of <?php ?> tags and 'extras'. 15%

For this assignment, what I want you to do is CREATE your own Github Repository for me to fork.