asgn03 Static methods and properties

Objectives

- Modify existing code to
    - Add static properties
    - Add static methods

- Use the ternary operator instead of if...esle
- Use Git and GitHub without explicit instructions.

Videos

Watch chapter 5 - Static Properties and Methods from the PHP: Object-oriented programming series.

- Create a folder called static-challenge in your web250 folder.
- Complete the static references challenge. The challenge will prepare you for the next part of the assignment.
- It is fine to use the solution to help you better understand the problem.

Setup

- Download and unzip the file **asgn03-static** from Moodle and place them into your **web250** folder.
- Use this code as your starter code for this assignment.
- I have separated the class, **Bird.php**, from **index.php**. This file differs from the screencast, but I think it makes the code easier to read. You will need to modify both of these files.

Git (review)

I added a link to the git tutorials from Laracasts called Git Me Some Version Control. Review these *as needed* for this and future assignments. This tutorials, like the other older tutorials on Laracasts is free.

Your are required to

- Stage new files (add)
- Commit when you need to
- Create a branch called dev
- Checkout to the branch dev.
- Merge completed code from the asgn03-static branch to the Main branch
- Push the contents of the Main branch to GitHub

Command Line or Not

I still prefer to use the command line. You can use it in VS Code or use a terminal. Git is part of VS Code, and there are extensions available.

You may choose to use another GUI for Git. A popular tool is Git Tower. It does not matter which route you take; what is essential is that you use git for version control.

Code

Modify the code to meet the following requirements.

Bird Class

Create the following.

- A static property named `instatnce_count`.
- A static property named `egg_num` and set its default to zero
- a static method named `create()` that creates a new instance of a Bird object.

Ternary operator

Modify the following method (also located in the Bird.php file), so it uses the PHP ternary operator.

```
function can_fly() {
if ( static::$flying == "yes" ) {
    $flying_string = "bird can fly";
    } else {
    $flying_string = " cannot fly and it stuck on the ground";
    }
    return $flying_string;
}
```

Flycatcher subclass

Add a static property named egg*num that overrides the Bird class egg*num by adding the string "3-4, sometimes 5."

Git

Stage and commit.

Index.php

- Display the `instance_count` for the Bird, Yellow-bellied Flycatcher, and Kiwi before using the `create()` method.
- Create new instances of Bird, Yellow-bellied Flycatcher, and Kiwi
- Display the `instance_count` for the Bird, Yellow-bellied Flycatcher, and Kiwi after using the `create()` method.

Git and GitHub

You should have completed all of your code at this point.

While you are on the dev branch

- Stage and commit your files
- Checkout to the main branch
- Merge main with your dev branch
- Push your code to GitHub

Note: You may get a warning that your local version of Git is out of sync with your Origin (online version). This happens when your local git files are out of sync with your online repo. Follow the messages that Git provides to fix the problem.

Post to the class Slack `#general` channel if you are having difficulty.

Submit Your Work

Submit a text file named **asgn03-yourLastName-YourFirstName.txt** in Moodle.

Next Week

We will start using online webhosting with a domain name next week. You may continue to use the one you created for WEB182.

Resources

Audubon