asgn01 Classes

Objectives

- Set up local environment
- Create a class
- Create class variables
- Create class methods
- Read class diagrams
- Use version control (Git) locally

The Assignment

There are four parts to this assignment.

1. Complete the bike **challenge**. I realize he provides an answer. Using git is optional for the bike challenge.
2. Complete the bird class exercise below. *Use git for this part of the assignment*.
3. Submit your `git log` in the comments section in Moodle
4. Submit your code in Moodle. We will push code to GitHub in an upcoming exercise. Zip the `asgn01` folder and name it `yourLastName-asgn01`.
5. Do not upload anytning to your web host yet.

Introduction

In the class, you are going to learn OOP (Object Oriented Programming) using PHP. One of the values of OOP is code reusability. Reusability is incredibly useful as your programs grow in size and complexity.

There are a lot of objectives in this assignment, but they are relatively small. The purpose is to set the foundation for this course.

Folder structure

You may continue to use the same local setup that you had in WEB 182 using XAMPP/MAMP/etc.

Windows

```
c:\xampp\htdocs\web250
```

Mac

```
/Applications/MAMP/htdocs/web250
```

Create the following folder structure. This will keep the tutorial exercises and extra coding separate

```
web250
    asgn01
        bike-challenge
        bird-class
```

Bike Challenge

Watch the LinkedIn/Learning series titled PHP: Object-Oriented Programming by Kevin Skoglund. In addition to the tutorials, I will add a programming assignment to help reinforce the concepts.

For this assignment watch the Introduction and

1. Introduction (Make sure to download the exercise material).
2. Overview and Project Setup
3. Object Basics

Complete the **Challenge** at the end of Object Basics. It will serve as a template for the assignment. Mr. Skoglund provides a solution but give it your best shot without burning up too much time. Or, watch his solution then go back and try and code the challenge without referencing it. Challenge yourself.

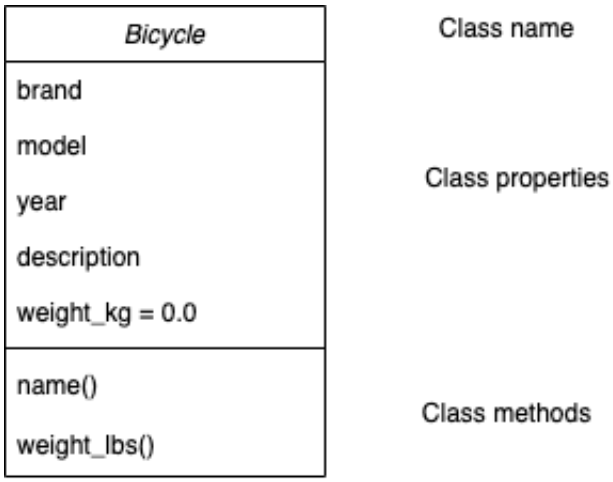
UML - Unified Modelling Language

Class Diagrams

UML is a diagramming system that is comprised of many different modelling techniques. We will use it for class diagrams. Just this bit will help you understand how classes work. You can think of the class diagrams as a visual algorithm. They are incredibly helpful in organizing your thoughts. We will not use UML for anything other than the class diagrams.

Here is a class diagram for the **Challenge**. Note that

- The class name is a singular noun and the first letter is capitalized. If the class name is more than one word, then use CamelCase.
- The properties (attributes) are next. You can include a default value.
- The bottom section displays the methods (oop speak); also called functions.



NOTE: The author's code is dated. He is using HTML4 and also uses underscores in his file names. Here is how we will handle this.

For his tutorials, continue using the same syntax. It is just easier and you are trying to learn the concepts.

When we break out coding assignments, use the more recent styles. There is a link to PHP styles in Moodle.

Git

Version control is an important part of this class. I will be using the command line, but most editors now incorporate a GUI for Git. Personally I think it is best to start with the command line before moving the GUI. That way, if something goes sideways, you will understand the foundation behind the GUI. You are welcome to use either as long as you use git throughout the term.

Side note: Employers expect you to know git!

Watch the video on git in Moodle

Windows

Use GitBASH or Powershell

Mac

Use the terminal (or iTerm2) or you can use the terminal built into VSCode.

Git Steps

If you run into an git error stating that you need to configure git, then open your terminal and do the following, using your name and email address

```
git config --global user.name "John Doe"
git config --global user.email johndoe@example.com
```

Once you have loaded up your **oop-sandbox** with the required files. The git video will walk you through all of these steps.

Work through the sandbox part of the video. Some of the functions he uses seem cryptic at this point, but they will come into play later.

Production Setup

You do not need to upload anything to your web host for this assignment. We will get to that soon.

Bird Class

Create a file named **index.php**. In the near future we will use class libraries but to get started we will follow the **challenge** example.

Inside the **index.php** follow do the following:

Create the following

- A class named Bird

- The following class properties
    - `commonName`
    - `habitat`
    - `nestPlacement`
    - `clutchSize` (number of eggs)
- Class methods
    - `birdSong`

Bird instances

Create a new instance of Bird called `$bird1`

Use this content. I have purposely left out the syntax

```
Properties
_____________________________
commonName = Eastern Towee
food = seeds, fruits, insects, spiders
nestPlacement = Ground
clutchSize = 2 - 6 eggs
conservationLevel = Low

Methods
_____________________________
birdSong = drink-your-tea!
```

Create a second instance of Bird called `$bird2`

```
Properties
_____________________________
commonName = Indigo Bunting
food = small seeds, berries, buds, and insects
nestPlacement = fields and on the edges of woods, roadsides, and railroad
 rights-of-way.
clutchSize = 3 - 4 eggs
conservationLevel = Low

Methods
_____________________________
birdSong = what! what!
```

For both of these instance you will want to display the content.

Run your program and test it locally.

Git

Now that your programming is running correctly, it is time for some additional version control.

- Open your terminal
- Navigate to your **web250** directory. This is the directory where your **.git file is located.
- `git status` to see where you are at
- `git add` to put your files in the staging area.
- `git commit -m"Finished asgn01"`

Git Log

- Use `git log` to display your work.
- Take a screenshot of the log output.
- Name the screenshot **asgn01-git.png**

Turn in your work

- Zip up your **asgn01** folder that contains both your bike challenge and your bird class. Include the **asgn01-git.png** file. Name the zipped file `asgn01-yourLastName` and put it in Moodle.