
SERVER-SIDE WEB DEVELOPMENT

Lecture 1

TODAY'S TOPICS



- Professor Introduction
- Course Introduction
- Install Development Tools
- Review Git & GitHub
- PHP Basics
- Participation: Hybrid #1

PROFESSOR INTRODUCTION

MICHAEL EISENBRAUN

(eyes in brown)

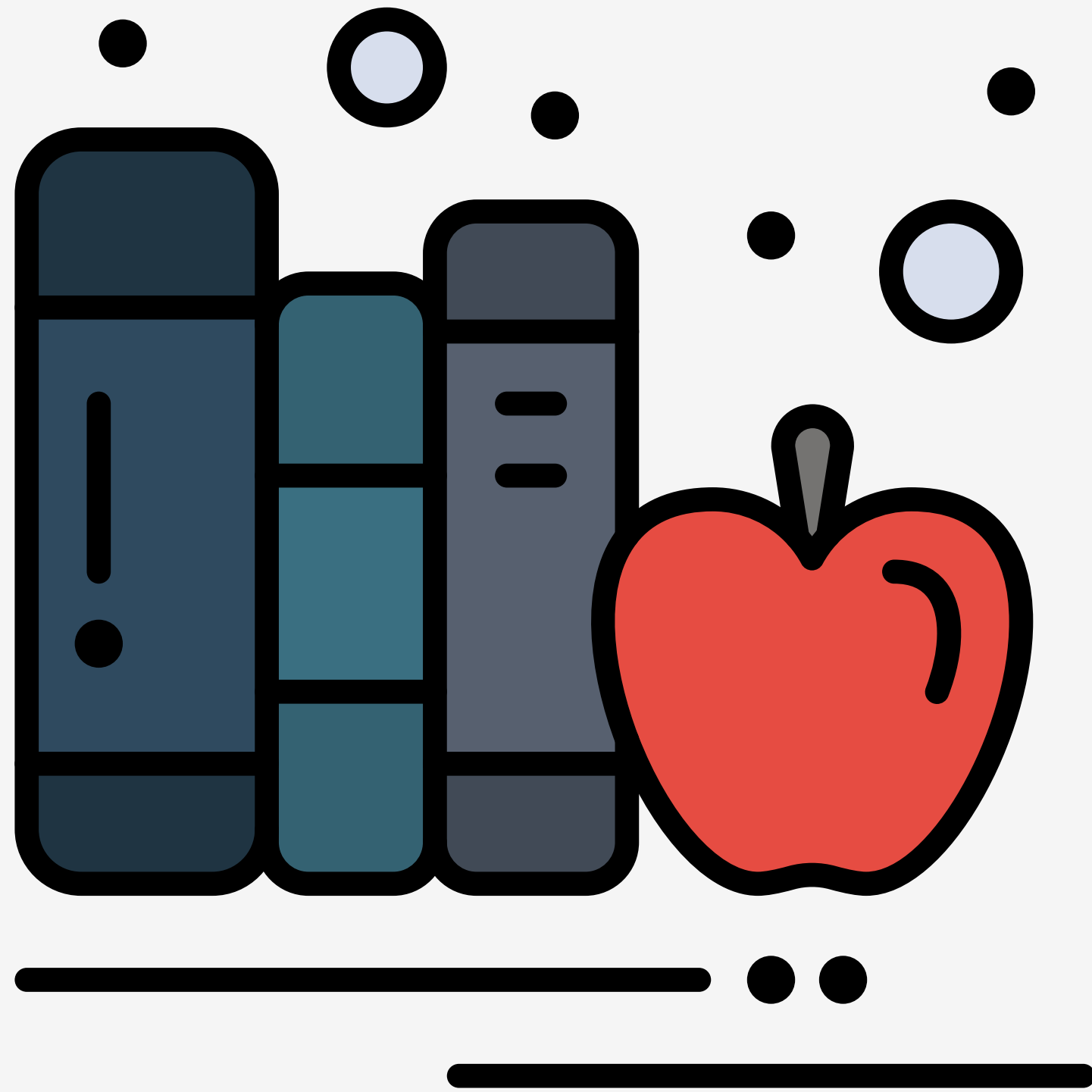
MICHAEL EISENBRAUN



- First web site in 1997
- Graduated from Iowa State in 2005
- First web develop job in 2006
- Immigrated to Canada in 2015
- Started teaching in 2016
- Wife and 3 Kids

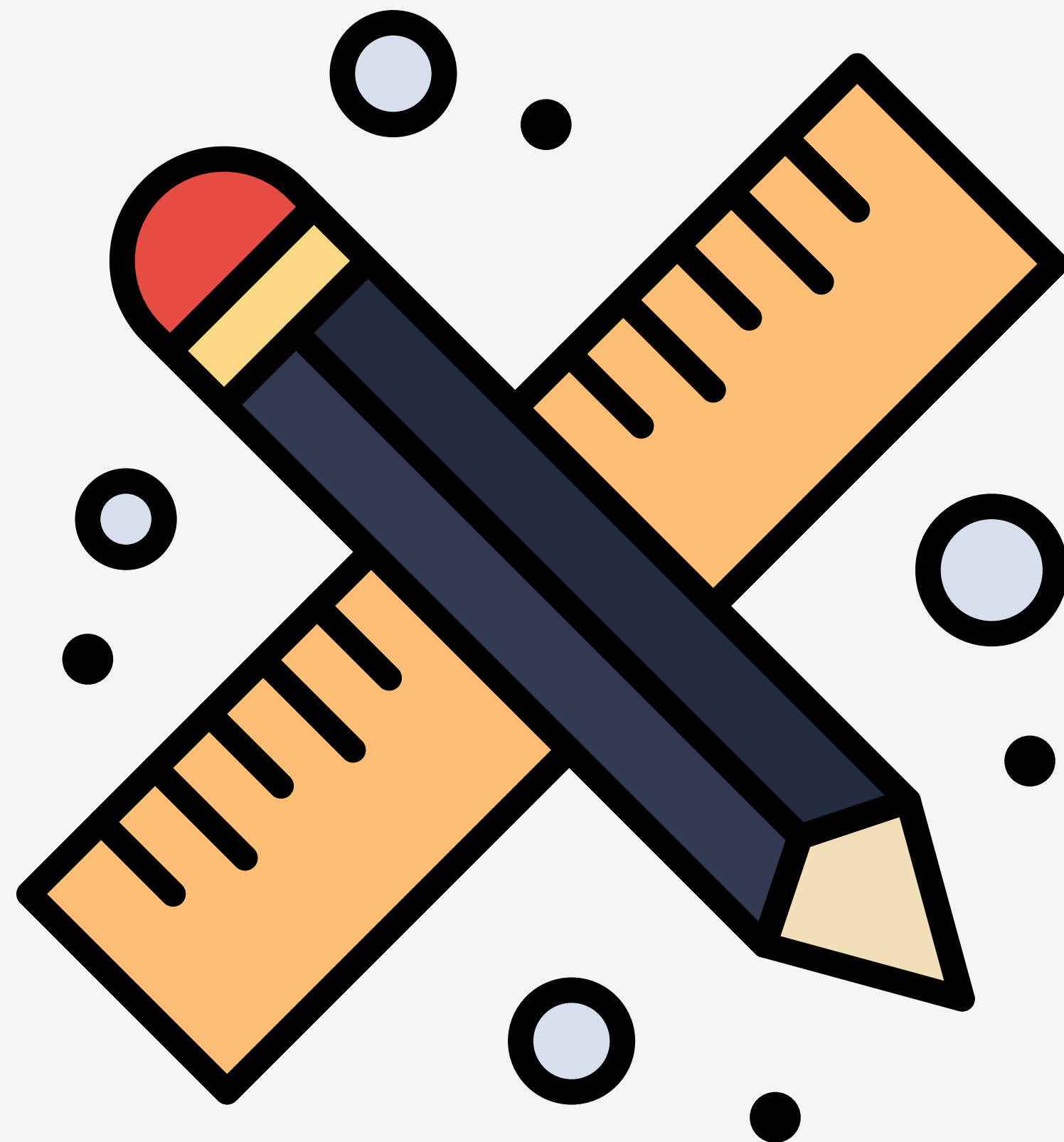
COURSE INTRODUCTION

COURSE TOPICS



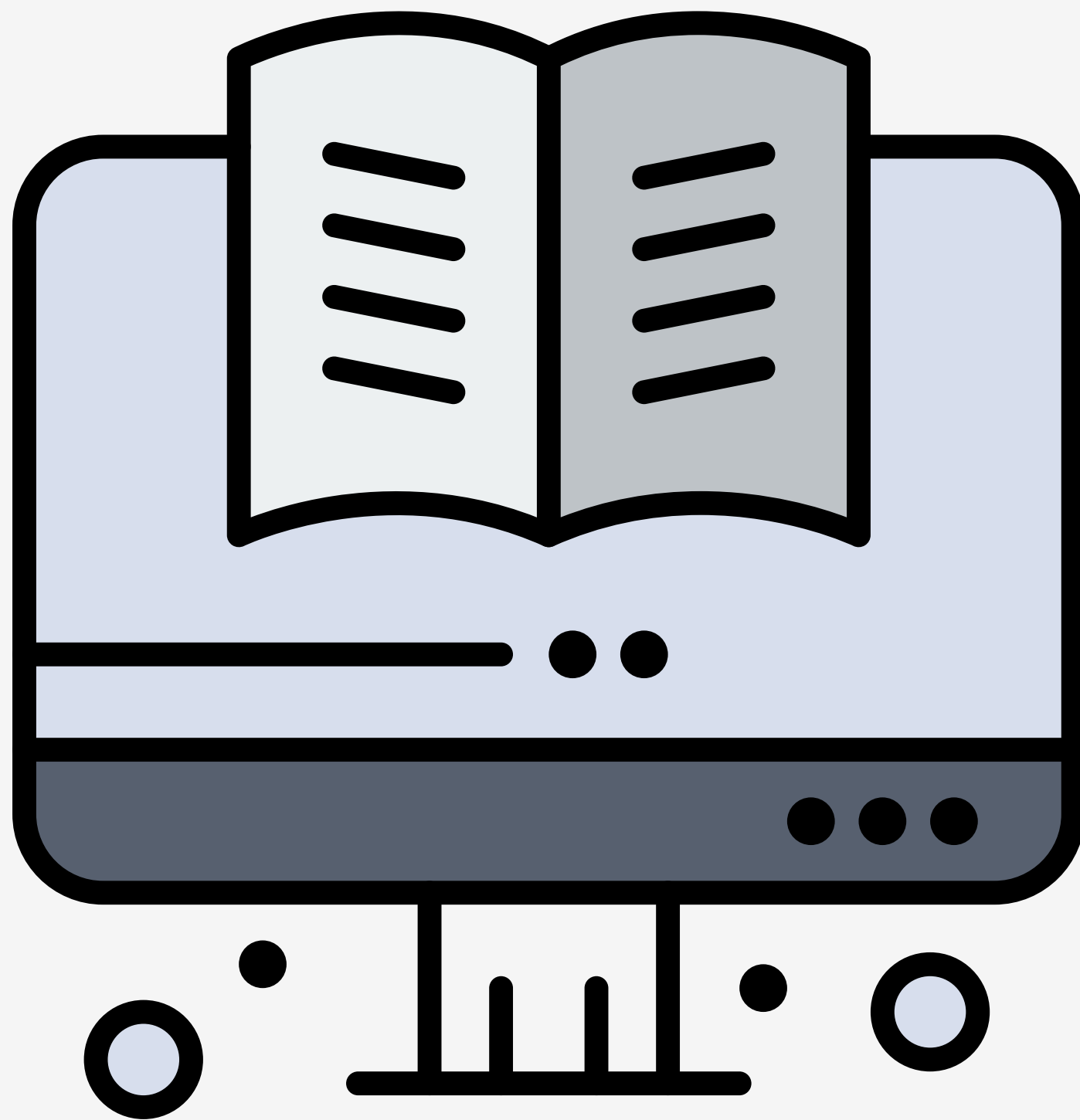
- PHP Basics
- PHP Requests
- PHP Includes
- Databases
- SQL Basics
- PHP Data Objects
- Laravel

ASSIGNMENTS



- 10 Participation (20%)
- 4 Exercises (30%)
- Midterm Project (25%)
- Final Project (25%)

COURSE CONTENT



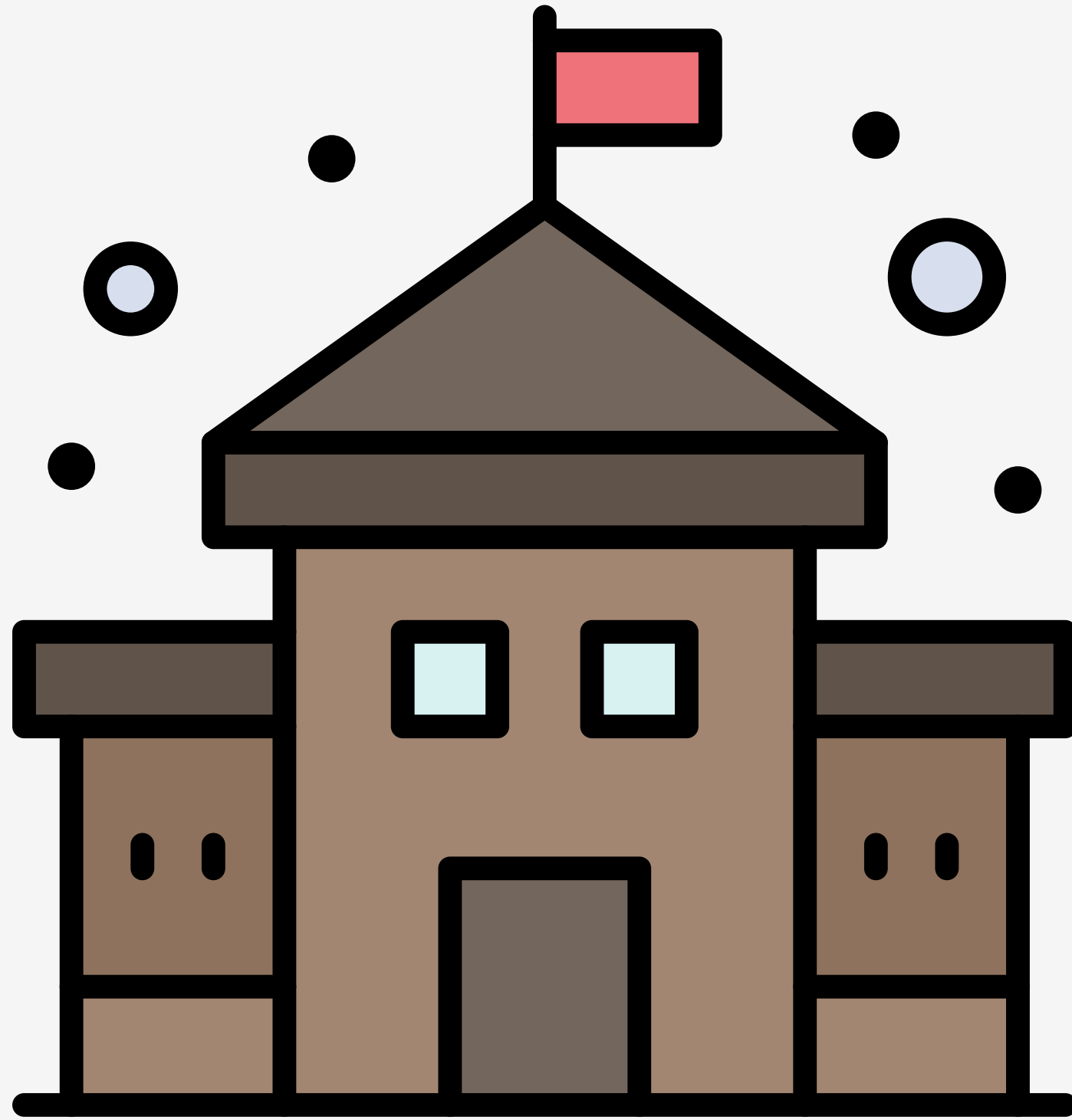
- IMDAC Website is use for content
- Spring & Winter 2020 recordings
- BrightSpace is grades
- GitHub Classroom for submission

COURSE STRUCTURE



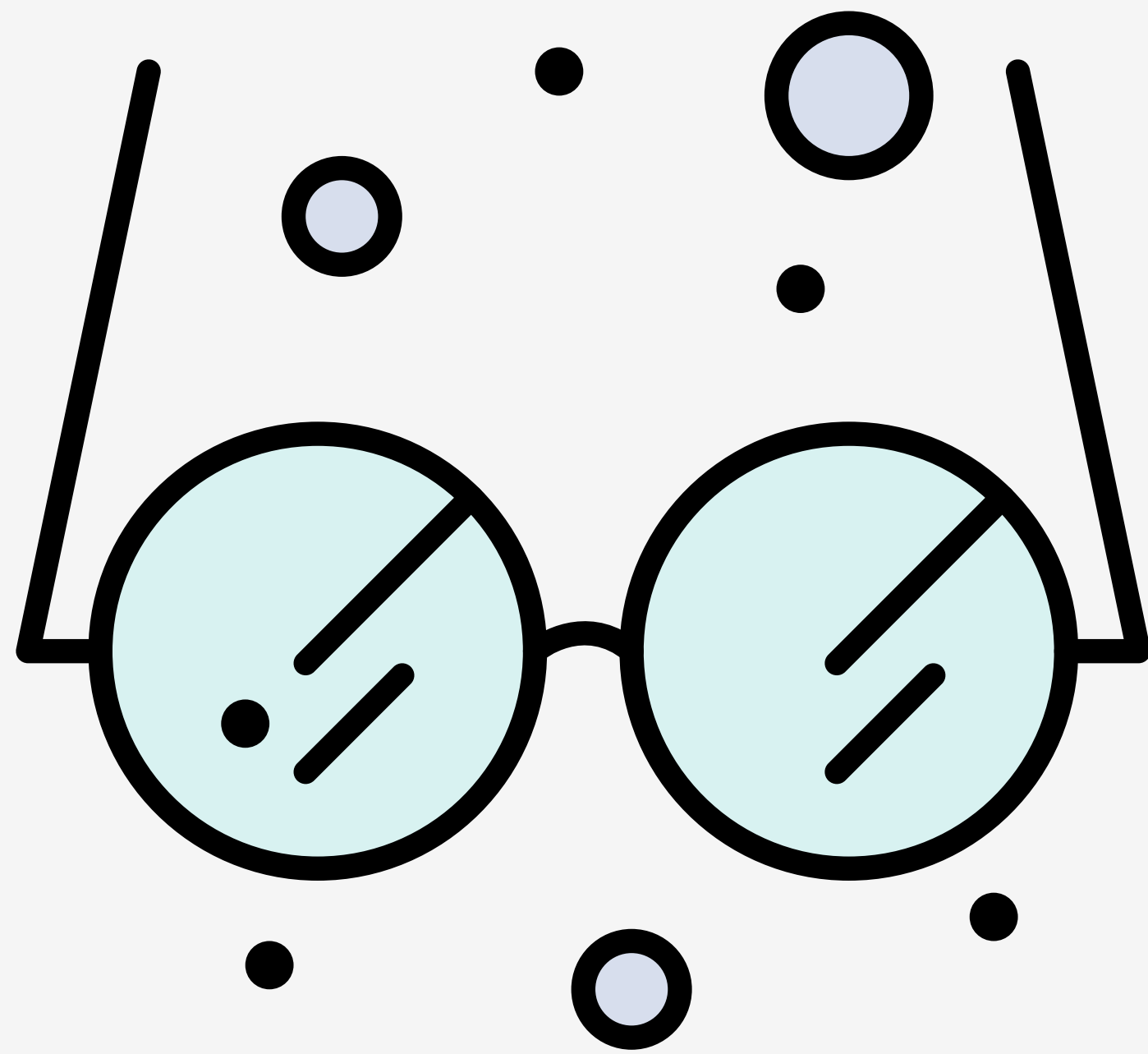
- 15 weeks
- Weekly live Zoom meets (recorded)
- Pre-recording examples
- Slides and recordings will be made available

STUDENT EXPECTATIONS



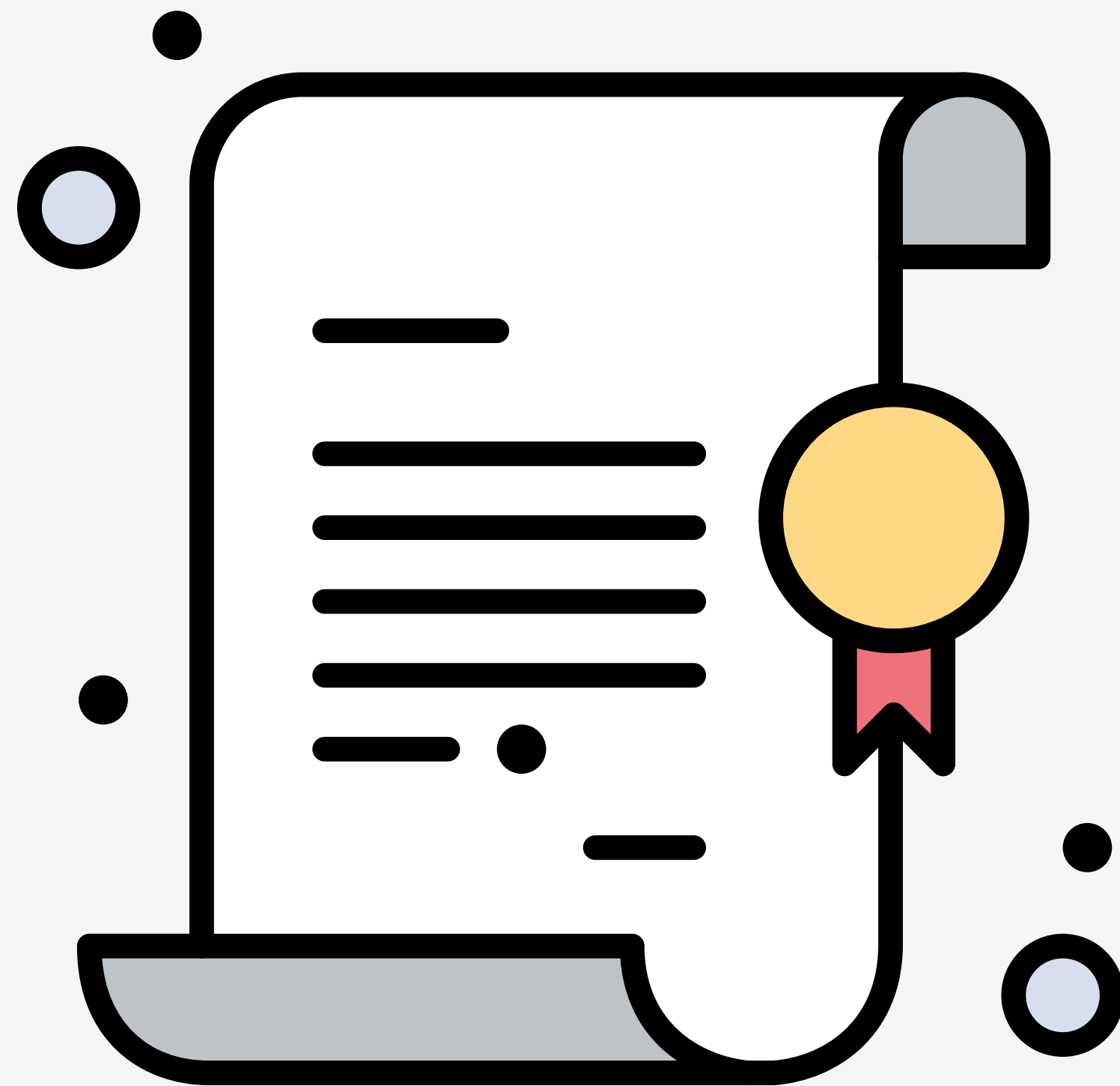
- Do the work
- Do your own work
- Don't be late
- Be respectful

PROFESSOR EXPECTATIONS



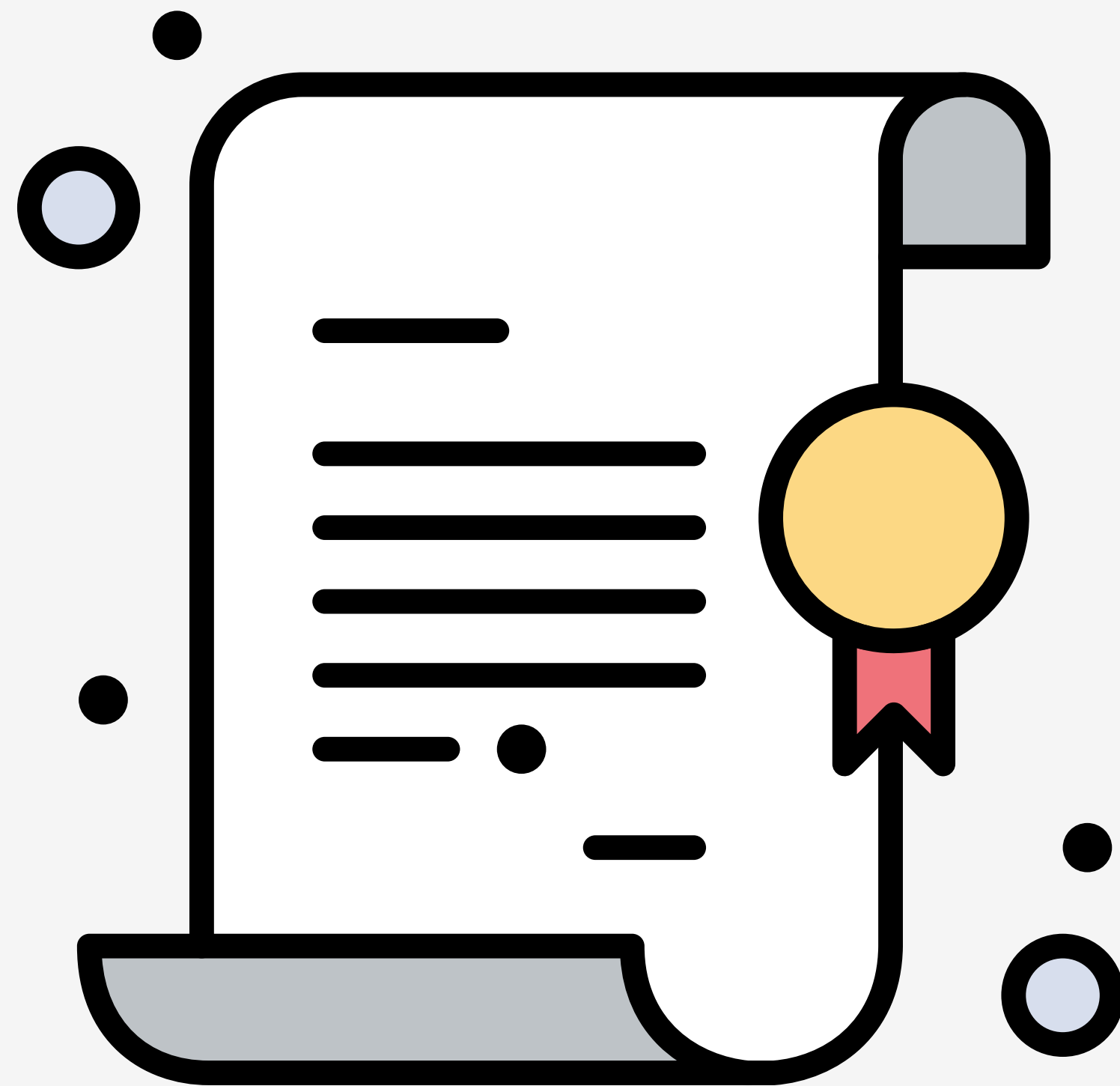
- Provide accurate and timely information
- Be flexible to the needs of the class
- Respond to emails within 24 hours
- Provide feedback within 1 week
- Fair and unbiased grading

PLAGIARISM & REFERENCING CODE



- Plagiarism is submitting someone else's work as your own **WITHOUT** proper reference
- Getting **ANY** code from online resource is considered plagiarism
- Sending or receiving code from a friend or classmate is considered plagiarism
- Working together on a project **MAY** fall under plagiarism

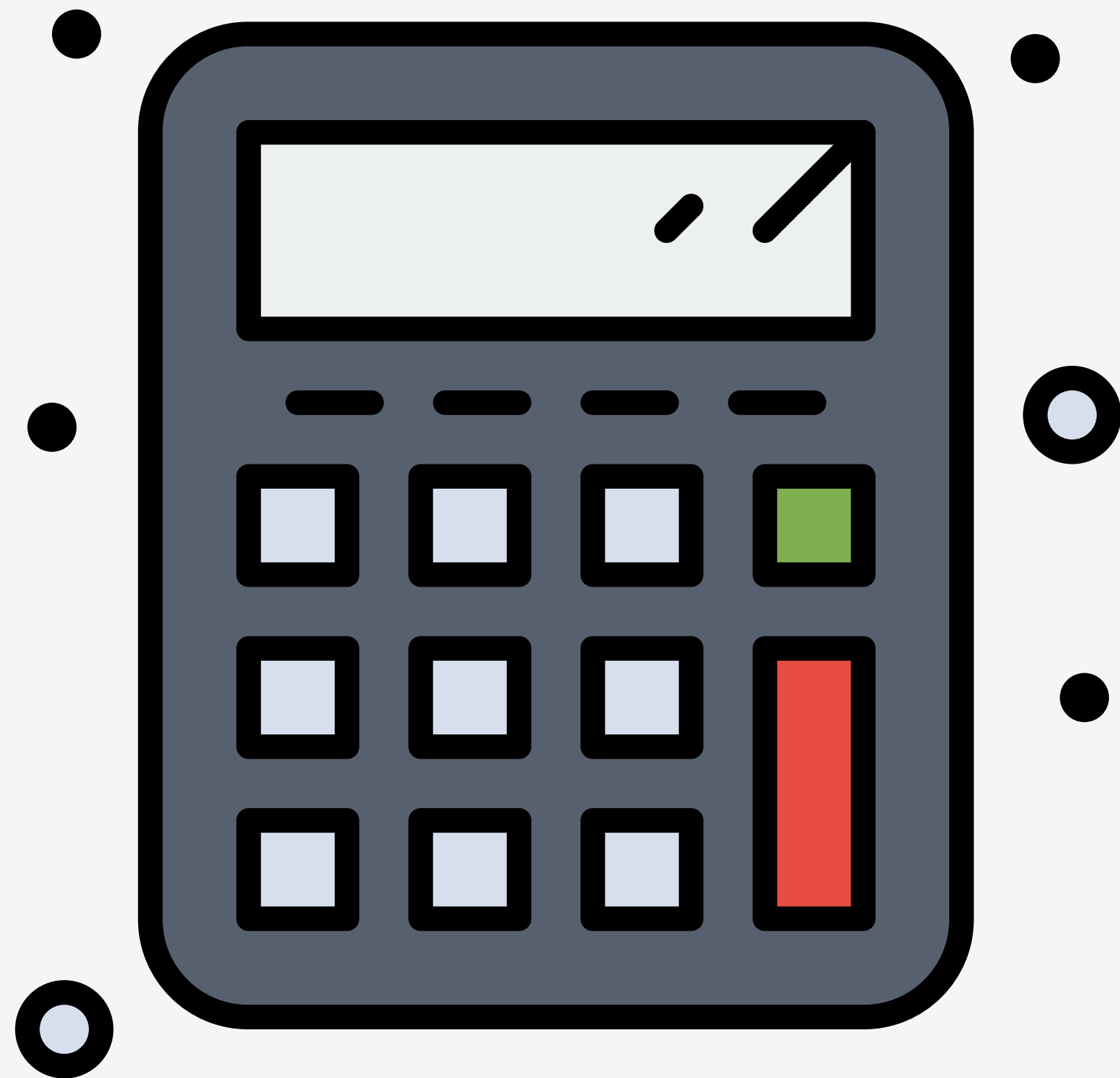
PLAGIARISM & REFERENCING CODE



- Any code that is not entirely your own should be referenced
- **ONLINE:** Include a description of what the code does and the source URL
- **PERSON:** Include a description of what the code does and the name of the person and when help was received
- **EXCEPTION:** Any code provided in class or in course content can be used without reference

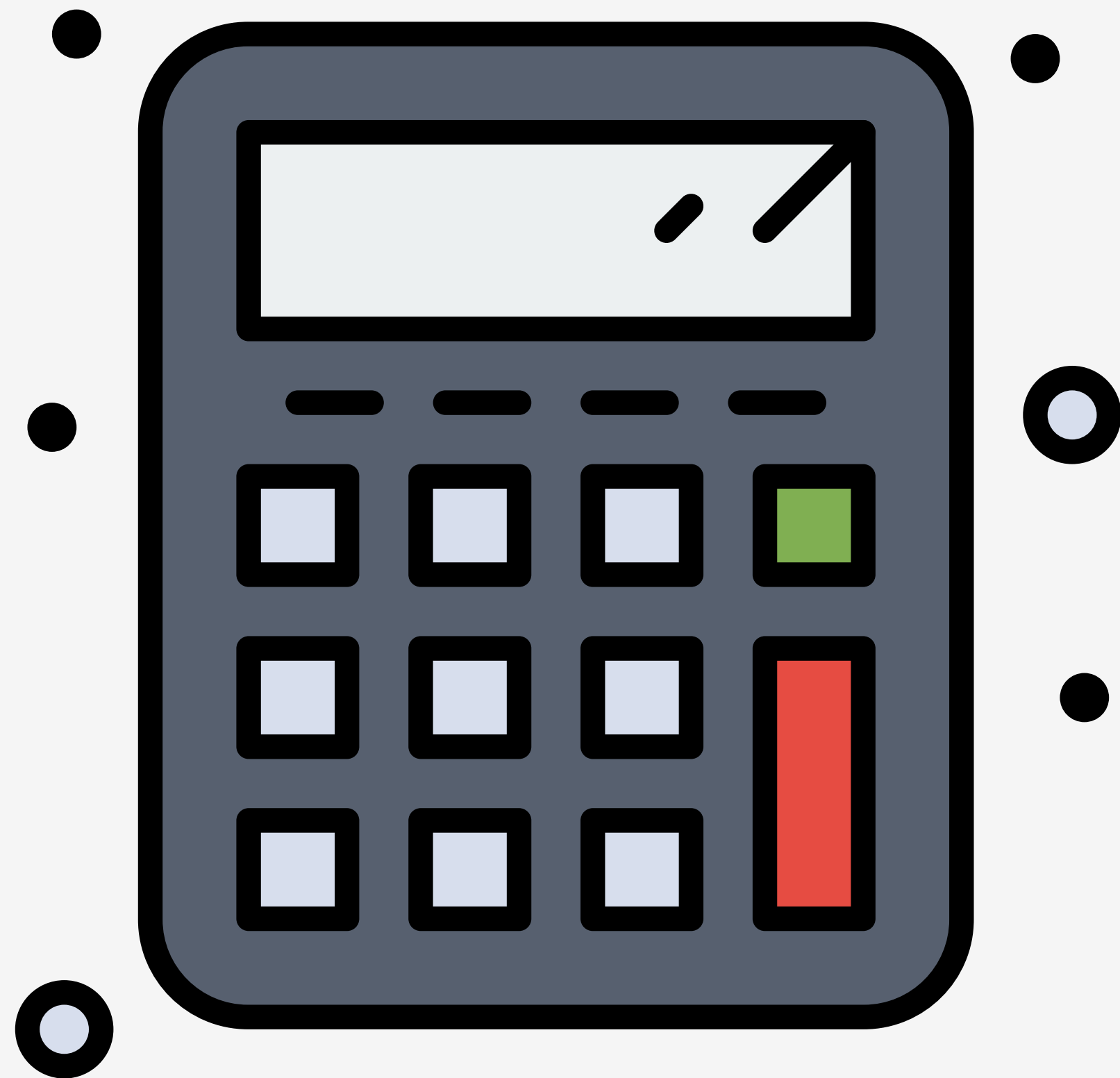
INSTALL DEVELOPMENT TOOLS

DEVELOPMENT TOOLS



- Git
- Node
- PHP
- MySQL
- Composer
- phpMyAdmin
- Xdebug

DEVELOPMENT TOOLS



- Laragon (Windows)
- Valet (macOS)

GIT & GITHUB

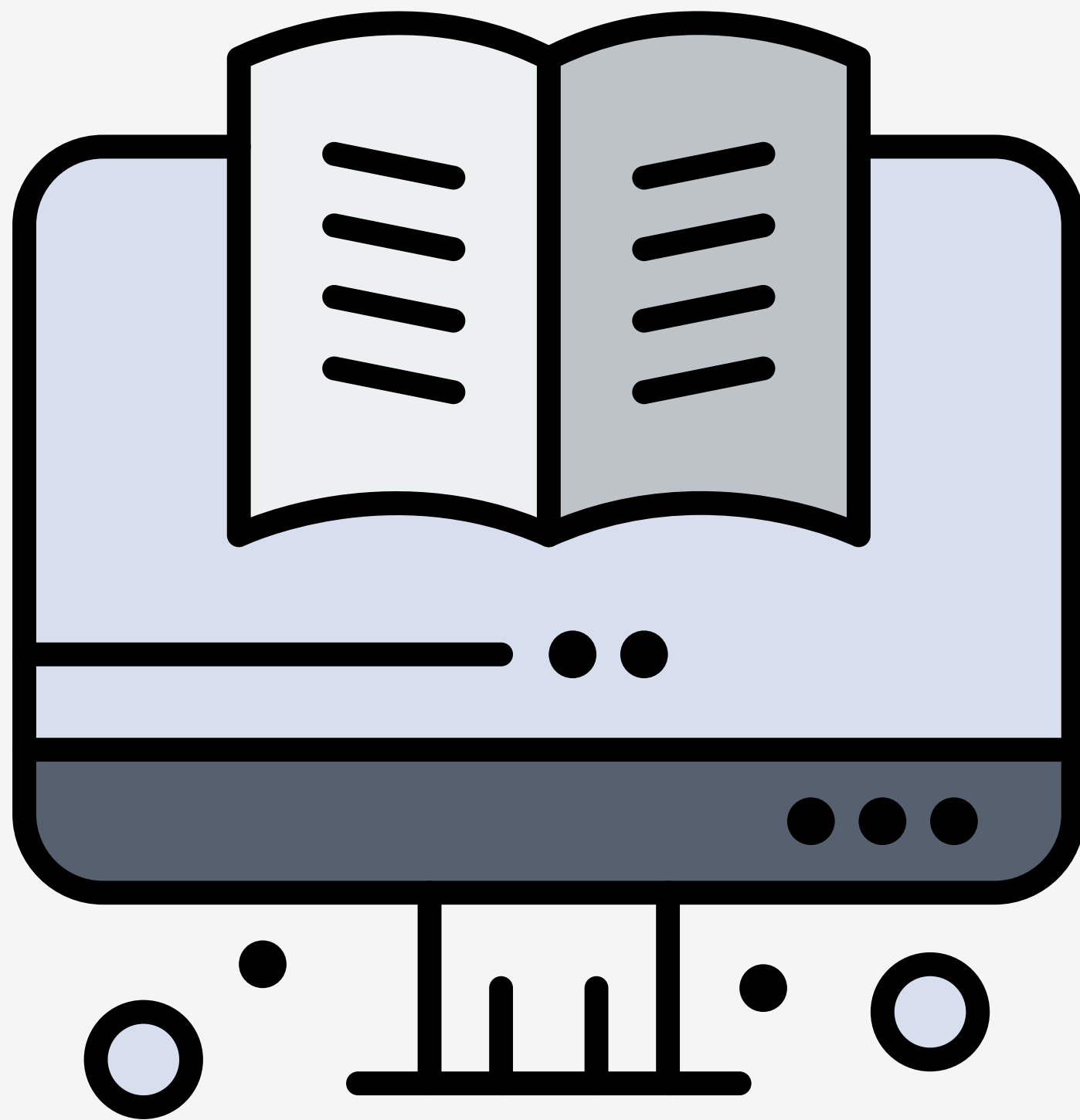
GIT & GITHUB



- GitHub Classroom
- Review Git & GitHub Basics

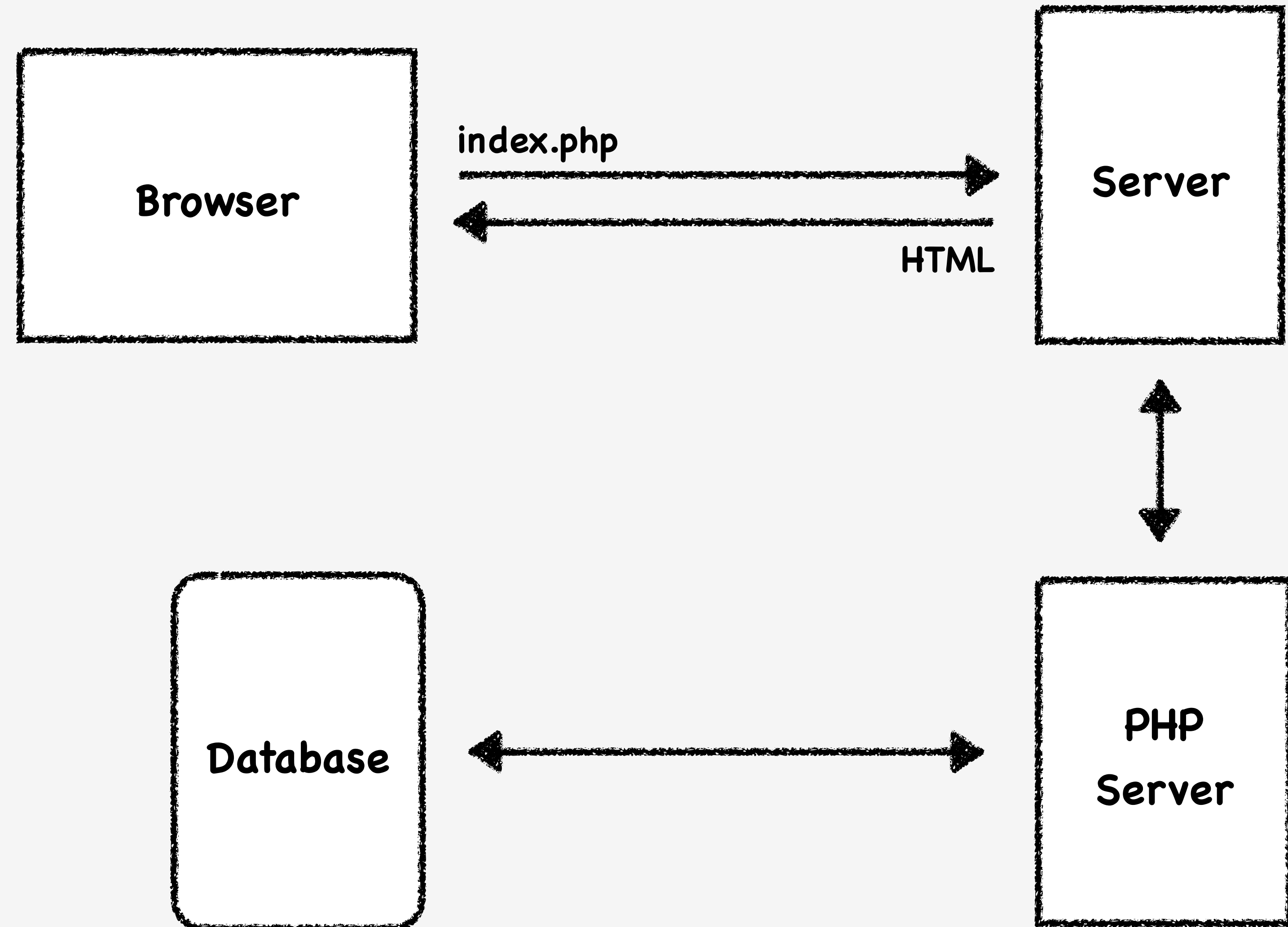
PHP BASICS

PHP HTML PREPROCESSOR

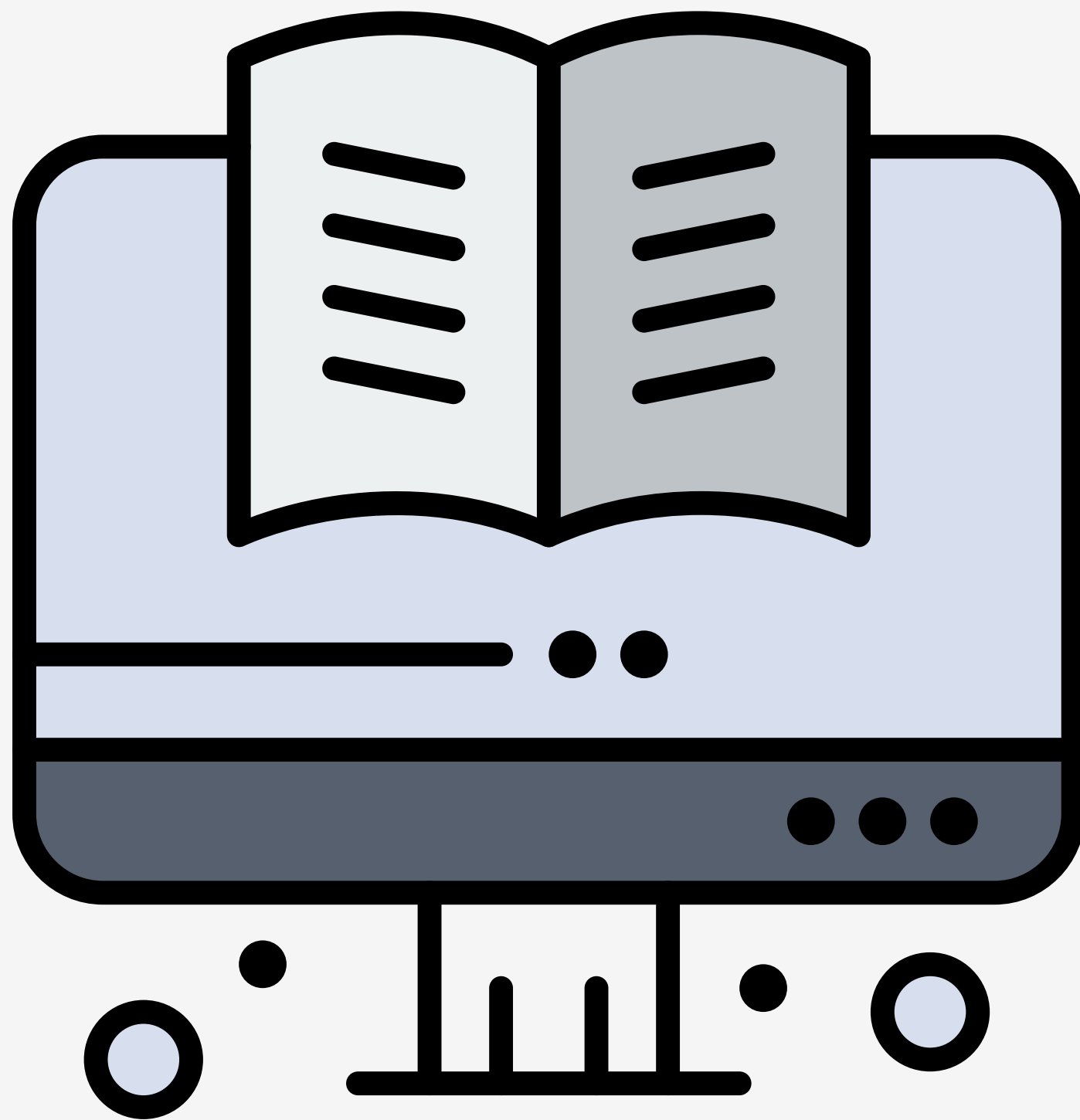


- General purpose scripting language
- HTML preprocessor
- Server-side language
- Cross-platform

HOW IS PHP PROCESSED



PHP SYNTAX



- PHP Extension (`.php`)
- PHP Tag (`<?php ?>`)
- Semicolons are required

PHP SYNTAX

```
<?php $title = "A Simple PHP File"; ?>
<html>
  <head>
    <title><?php echo $title; ?></title>
  </head>
  <body>
    <?php echo "Hello World"; ?>
  </body>
</html>
```

PHP ECHO

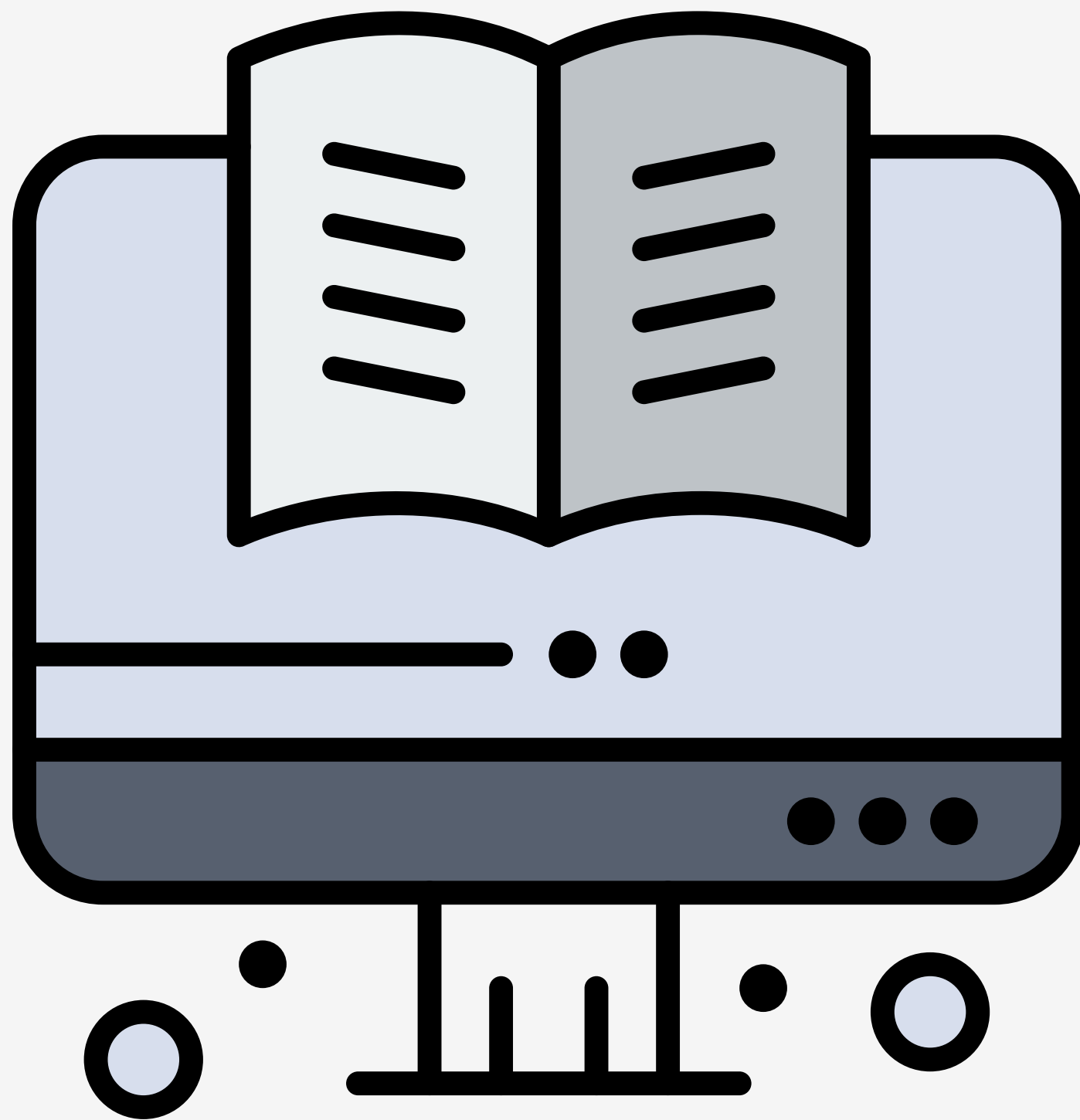


- Used to output one or more strings
- Primary to output PHP data as HTML

PHP ECHO

```
<?php $title = "A Simple PHP File"; ?>
<html>
  <head>
    <title><?php echo $title; ?></title>
  </head>
  <body>
    <?php echo "Hello World"; ?>
  </body>
</html>
```

PHP VARIABLES

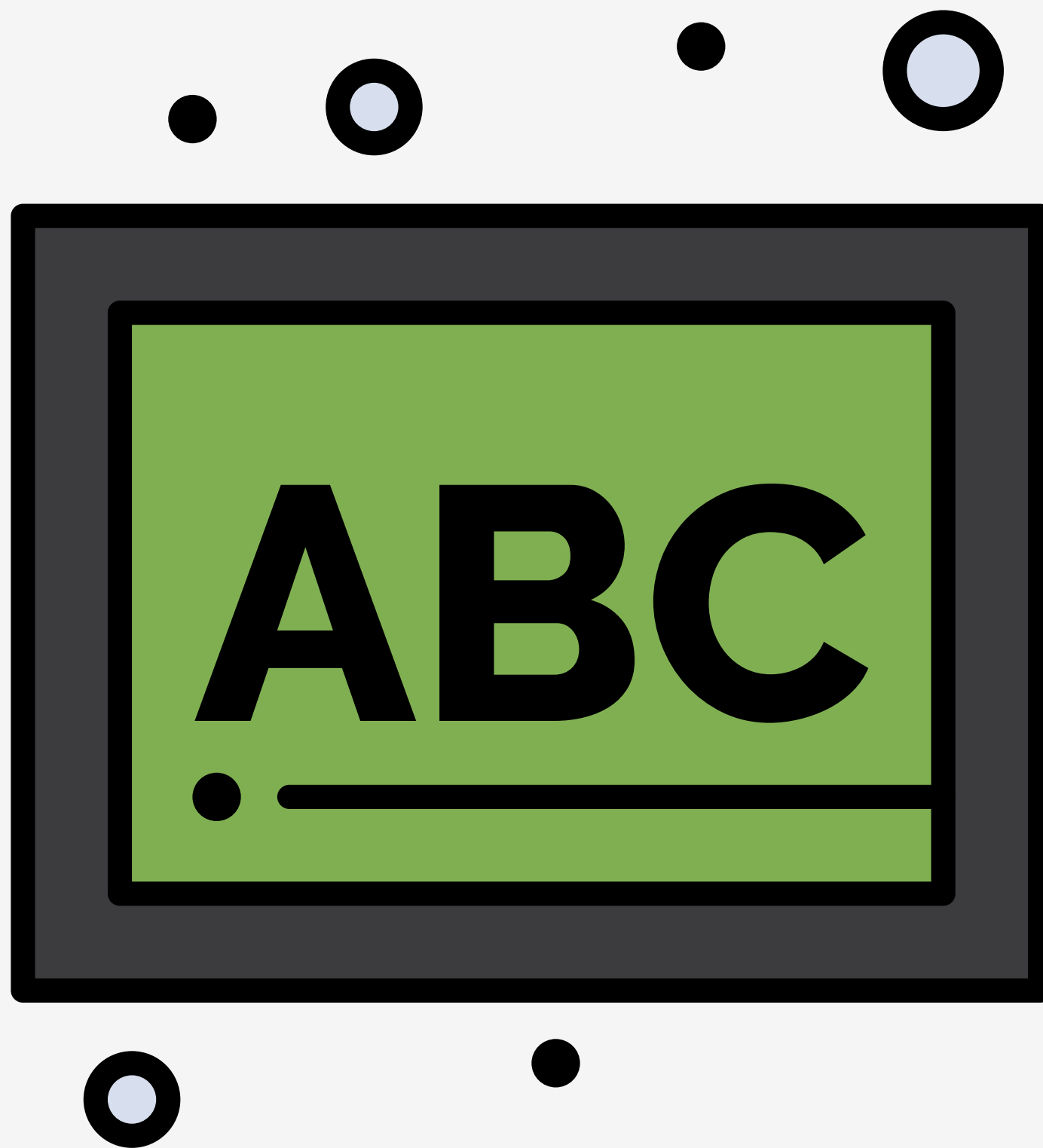


- Variables are used to hold data including strings, numbers, and arrays
- Variable names must start with a dollar sign (\$)
- Variable names can contains letters, numbers, underscores, or dashes and are case-sensitives
- No declaration statement

PHP VARIABLES

```
<?php
    // gives $num a value
    $num = 10;
    echo $num; // outputs 10
```

PHP STRINGS



- Strings can be created using **single quotes** or **double quotes**
- Use the dot (**.**) is used for **string concatenation**
- **Variable substitution** occurs when a variable is embedded into a string literal (***double quotes only***)

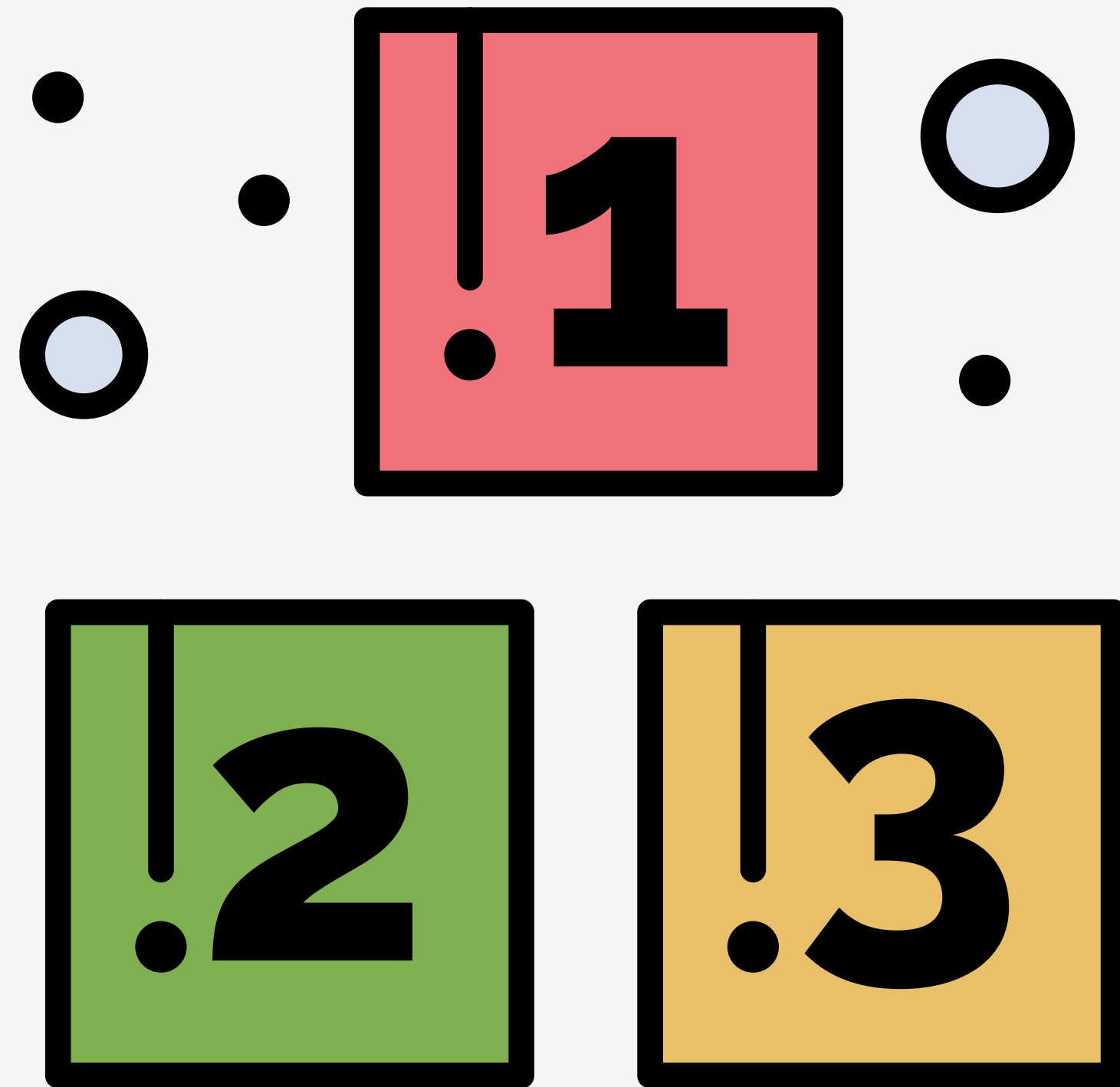
PHP STRINGS

```
<?php
    // setting the variables to strings
    $greeting = "Hello";
    $target = "World";

    // combine variables together with a string
    $phrase = $greeting . " " . $target;
    echo $phrase;

    // using variable substitution
    echo "{$phrase} Again!";
```

PHP ARRAYS



- An **array** is an indexed list of values
- Indexes start with **0**
- **Bracket notation** is used to retrieve values
- Values can be added, changed or removed from an array
- An array containing other arrays requires an additional notation

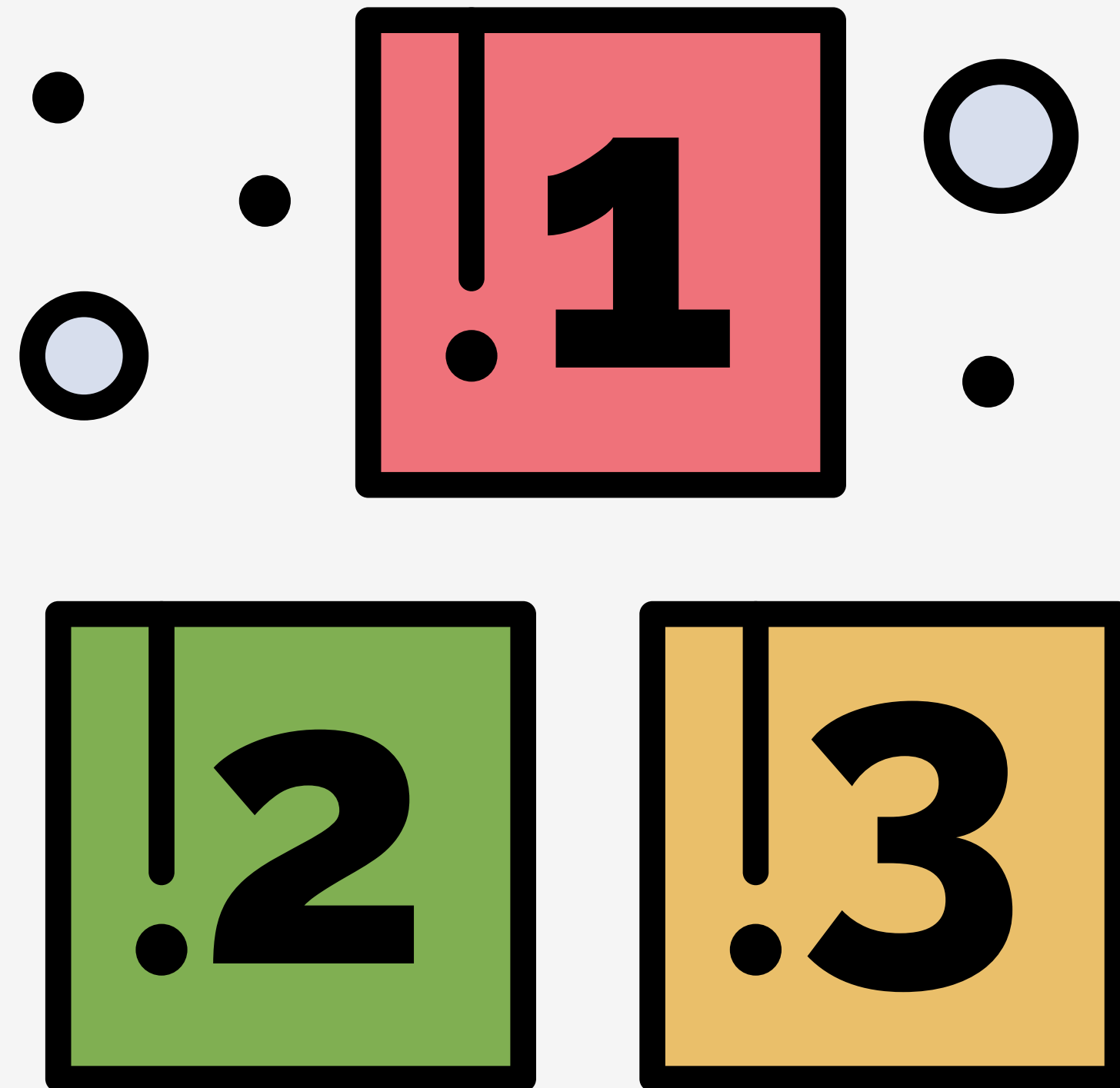
PHP ARRAYS

```
<?php
    $numbers = [4, 8, 15, 16, 23, 42];

    // getting the second item
    echo $numbers[1]; // 8

    // getting the first item
    echo $numbers[0]; // 4
```

PHP ASSOCIATIVE ARRAYS



- **Associative arrays** uses strings instead of numbers which serve as labels or keys
- Both **key** and **value** must be provided when defining an associative array
- The key is used to both retrieve and update the value

PHP ASSOC. ARRAYS

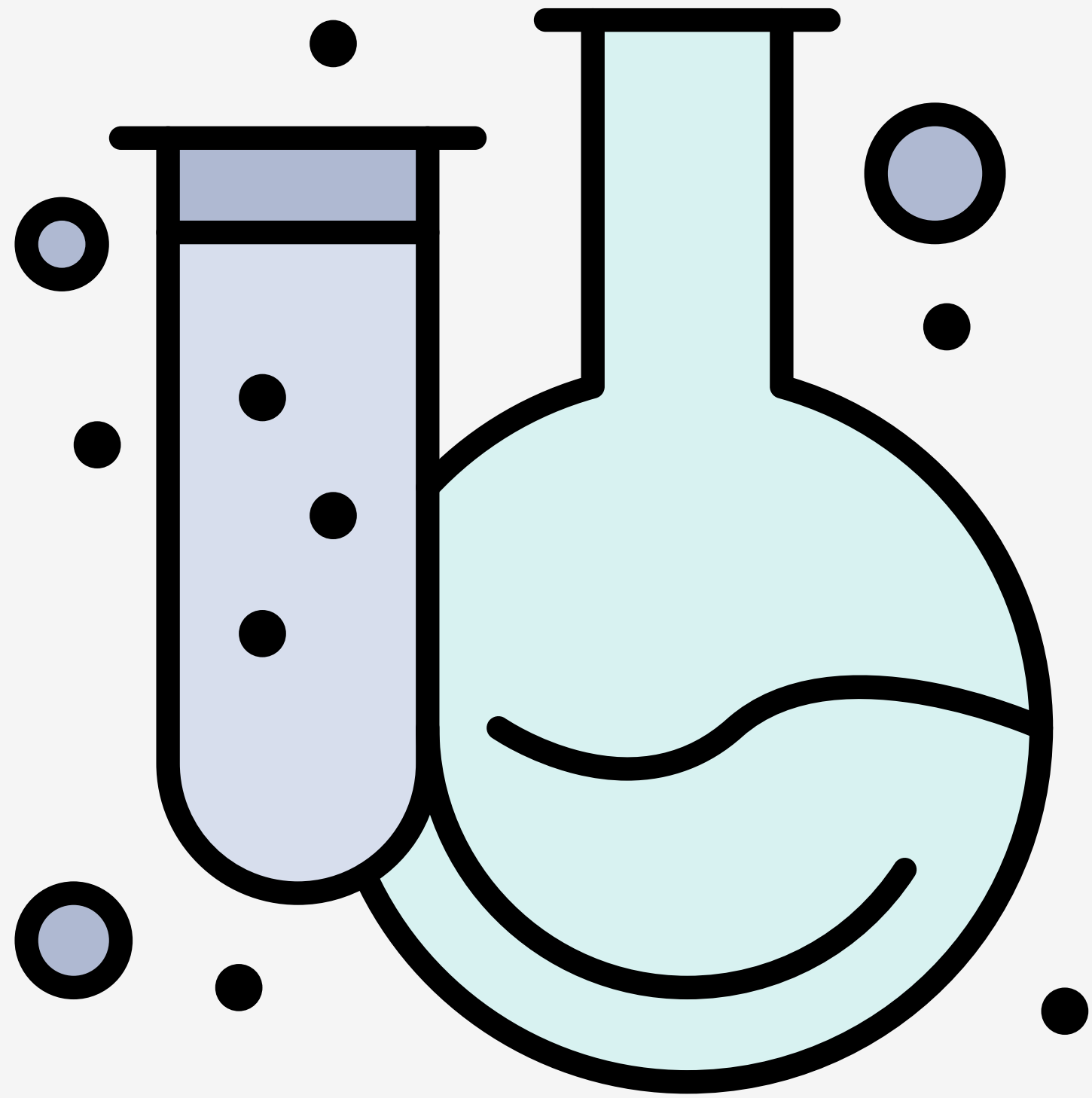
```
<?php
    // creating an assoc array
    $assoc = [
        "first_name" => "Michael",
        "last_name" => "Eisenbraun"
    ];

    // get the first name
    echo $assoc["first_name"]; // Michael

    // assign new value to first name
    echo $assoc["first_name"] = "Larry";

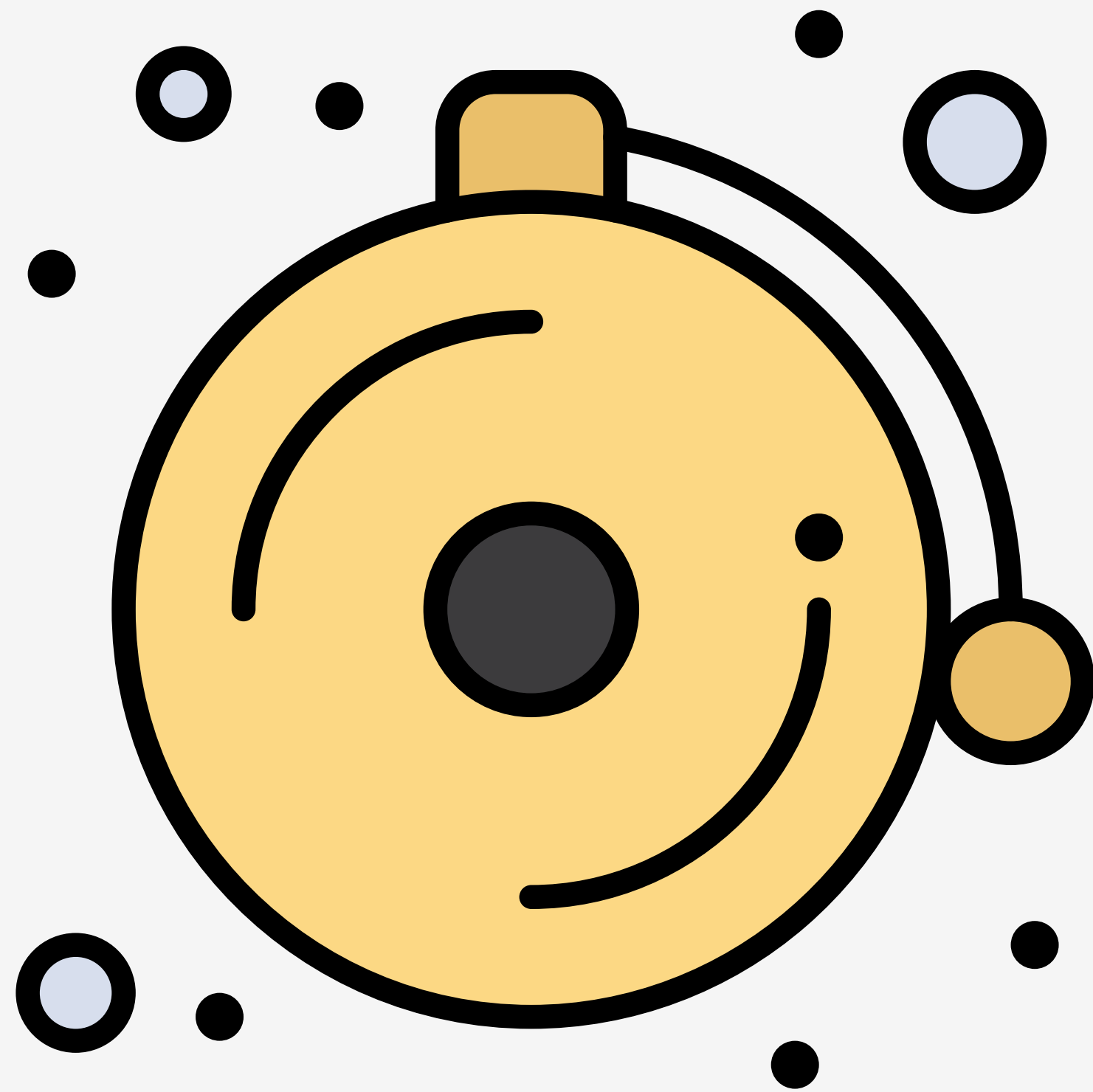
    // adding age to the array
    $assoc["age"] = 33;
```

HYBRID #1



- Watch any 10 episodes of **PHP Tips, Tricks, and Techniques** on LinkedIn Learning
- Write 1 to 2 sentences for each episode
- ***DUE:*** Thu. Oct. 1 @ 11:59 PM

NEXT TIME...



- PHP Control Flow
- **Participation:** Deck of Cards
- **Exercise:** Dominoes