**Exercise 02\_03\_01 – Step 1**

MusicalScale.php

<head>

<!--

Exercise 02\_03\_01

Author: Mark J. Buckler

Date: September 24, 2017

Filename: MusicalScale.php

-->

<title>Musical Scale</title>

<meta charset="UTF-8">

<meta name="viewport" content="initial-scale=1.0">

<script src="modernizr.custom.65897.js"></script>

</head>

<body>

<h2>Musical Scale</h2>

<?php

?>

</body>

</html>

------------------------------------

MusicalScale.php

<?php

$musicalScale = array("do", "re", "me", "fa", "so", "la",   
 "ti");

?>

------------------------------------

MusicalScale.php

$outputString = "The notes of the musical scale are: ";

foreach ($musicalScale as $currentNote) {

$outputString .= " " . $currentNote;

}

------------------------------------

MusicalScale.php

echo "<p>$outputString</p>";

**Exercise 02\_03\_01 – Step 2**

FormattedText.php

<h2>Formatted Text</h2>

<?php

$displayValue = 9.876;

?>

------------------------------------

FormattedText.php

$displayValue = 9.876;

echo "<pre>\n";

echo "Unformatted text line 1. ";

echo "Unformatted text line 2. ";

echo "$displayValue = $displayValue";

echo "</pre>\n";

**Exercise 02\_03\_01 – Step 3**

BooksAndAuthors.php

<h2>Books And Authors</h2>

<?php

?>

------------------------------------

BooksAndAuthors.php

<?php

$books = array("The Adventures of Huckleberry Finn",   
 "Nineteen Eighty-Four", "Alice's Adventures in   
 Wonderland", "The Cat in the Hat");

?>

------------------------------------

BooksAndAuthors.php

$authors = array("Mark Twain", "George Orwell", "Lewis   
 Carroll", "Dr. Seuss");

------------------------------------

BooksAndAuthors.php

$realNames = array("Samuel Clemens", "Eric Blair", "Charles   
 Dodson", "Theodor Geisel");

------------------------------------

BooksAndAuthors.php

for ($i = 0; $i < count($books); $i++) {

echo "<p>The real name of {$authors[$i]}, " . "the   
 author of \"{$books[$i]}\", " . "is   
 {$realNames[$i]}.</p>";

}

**Exercise 02\_03\_01 – Step 4**

BooksAndAuthors2.php

for ($i = 0; $i < count($books); $i++) {

echo "<p>The title \"{$books[$i]}\" contains " .   
 strlen($books[$i]) . " characters and " .   
 str\_word\_count($books[$i]) . " words.</p>";

}

**Exercise 02\_03\_01 – Step 5**

WordPlay.php

<?php

$startingText = "mAdAm, i'M aDaM.";

?>

------------------------------------

WordPlay.php

$startingText = "mAdAm, i'M aDaM.";

$uppercaseText = strtoupper($startingText);

$lowercaseText = strtolower($startingText);

echo "<p>$uppercaseText</p>\n";

echo "<p>$lowercaseText</p>\n";

------------------------------------

WordPlay.php

echo "<p>" . ucfirst($lowercaseText) . "</p>\n";

echo "<p>" . lcfirst($uppercaseText) . "</p>\n";

$workingText = ucwords($lowercaseText);

echo "<p>$workingText</p>\n";

**Exercise 02\_03\_01 – Step 6**

WordPlay2.php

echo "<p>" . md5($workingText) . "</p>\n";

echo "<p>" . substr($workingText, 0, 6) . "</p>\n";

echo "<p>" . substr($workingText, 7) . "</p>\n";

echo "<p>" . strrev($workingText) . "</p>\n";

echo "<p>" . str\_shuffle($workingText) . "</p>\n";

**Exercise 02\_03\_01 – Step 7**

PHPEmail.php

<?php

$emailAddresses = array("john.smith@php.test",

"mary.smith.mail@php.example",

"john.jones@php.invalid",

"alan.smithee@test",

"jsmith456@example.com",

"jsmith456@test",

"mjones@example",

"mjones@example.net",

"jane.a.doe@example.org");

?>

------------------------------------

PHPEmail.php

function validateAddress($address) {

if (strpos($address, '@') !== false && strpos($address,   
 '.') !== false) {

return true;

}

else {

return false;

}

}

------------------------------------

PHPEmail.php

foreach ($emailAddresses as $address) {

if (validateAddress($address) === false) {

echo "<p>The e-mail address <em>$address</em> does   
 not appear to be valid.</p>";

}

}

**Exercise 02\_03\_01 – Step 8**

Presidents.php

$presidents = array("George Washington",

"John Adams",

"Thomas Jefferson",

"James Madison",

"James Monroe");

------------------------------------

Presidents.php

$yearsInOffice = array("1789 to 1797",

"1797 to 1801",

"1801 to 1809",

"1809 to 1817",

"1817 to 1825");

------------------------------------

Presidents.php

$outputTemplate = "<p>President [NAME] served from   
 [TERM]</p>\n";

------------------------------------

Presidents.php

foreach ($presidents as $sequence => $name) {

$tempString = str\_replace("[NAME]", $name,   
 $outputTemplate);

echo $tempString;

}

------------------------------------

Presidents.php

foreach ($presidents as $sequence => $name) {

$tempString = str\_replace("[NAME]", $name,   
 $outputTemplate);

$outputString = str\_replace("[TERM]",   
 $yearsInOffice[$sequence], $tempString);

echo $outputString;

}

**Exercise 02\_03\_01 – Step 9**

Presidents2.php

$presidents = "George Washington;John Adams;Thomas   
 Jefferson;James Madison;James Monroe";

------------------------------------

Presidents2.php

$thisPresident = strtok($presidents, ";");

------------------------------------

Presidents2.php

while ($thisPresident != NULL) {

echo "$thisPresident<br>";

$thisPresident = strtok(";");

}

------------------------------------

Presidents2.php

$thisPresident = strtok($presidents, "; ");

while ($thisPresident != NULL) {

echo "$thisPresident<br>";

$thisPresident = strtok("; ");

}

**Exercise 02\_03\_01 – Step 10**

PasswordFields.php

$record = "jdoe:8W4dSO3a39Yk2:1463:24:John   
 Doe:/home/jdoe:/bin/bash";

------------------------------------

PasswordFields.php

$passwordFields = array("login name",

"optional encrypted password",

"numerical user ID",

"numerical group ID",

"user name or comment field",

"user home directory",

"optional user command   
 interpreter");

------------------------------------

PasswordFields.php

$fieldIndex = 0;

$extraFields = 0;

$currField = strtok($record, ":");

while ($currField != NULL) {

if ($fieldIndex < count($passwordFields)) {

echo "<p>The {$passwordFields[$fieldIndex]} is   
 <em>$currField</em></p>\n";

}

else {

++$extraFields;

echo "<p>Extra field # $extraFields is   
 <em>$currField</em></p>\n";

}

$currField = strtok(":");

++$fieldIndex;

}

**Exercise 02\_03\_01 – Step 11**

PasswordFields2.php

$fields = explode(":", $record);

------------------------------------

PasswordFields2.php

foreach ($fields as $fieldIndex => $fieldValue) {

if ($fieldIndex < count($passwordFields)) {

echo "<p>The {$passwordFields[$fieldIndex]} is   
 <em>$fieldValue</em></p>\n";

}

else {

++$extraFields;

echo "<p>Extra field # $extraFields is   
 <em>$fieldValue</em></p>\n";

}

}

**Exercise 02\_03\_01 – Step 12**

PHPEmail2.php

function sortAddresses($addresses) {

$sortedAddresses = array();

$ilimit = count($addresses)-1;

$jlimit = count($addresses);

for ($i = 0; $i < $ilimit; $i++) {

$currentAddress = $addresses[$i];

for ($j = $i + 1; $j < $jlimit; $j++) {

if ($currentAddress > $addresses[$j]) {

$tempVal = $addresses[$j];

$addresses[$j] = $currentAddress;

$currentAddress = $tempVal;

}

}

$sortedAddresses[] = $currentAddress;

}

return($sortedAddresses);

}

------------------------------------

PHPEmail2.php

$sortedAddresses = sortAddresses($emailAddresses);

$sortedAddressList = implode(", ", $sortedAddresses);

echo "<p>Sorted Addresses: $sortedAddressList</p>\n";

------------------------------------

PHPEmail2.php

foreach ($sortedAddresses as $address) {

**Exercise 02\_03\_01 – Step 13**

PHPEmail3.php

if (strcasecmp($currentAddress, $addresses[$j]) > 0) {

**Exercise 02\_03\_01 – Step 14**

PHPEmail4.php

if (preg\_match("/^[\w-]+(\.[\w-]+)\*@[\w-]+(\.[w-]+)\*(\.[A-Za-z]{2,})$/i" ,$address) == 1) {

**Project 02\_03\_01**

**Part 1**

ValidateCreditCard.php

<head>

<!--

Exercise 02\_03\_01

Author: Mark J. Buckler

Date: October 3, 2017

Filename: ValidateCreditCard.php

-->

<title>Validate Credit Card</title>

<meta charset="UTF-8">

<meta name="viewport" content="initial-scale=1.0">

<script src="modernizr.custom.65897.js"></script>

</head>

<body>

<h2>Validate Credit Card</h2>

<?php

?>

</body>

</html>

------------------------------------

ValidateCreditCard.php

<?php

$creditCard = array("",

"8910-1234-5678-6543",

"OOOO-9123-4567-0123");

?>

ValidateCreditCard.php

foreach ($creditCard as $indexNumber => $cardNumber) {

if (empty($cardNumber)) {

echo "<p>Credit Card Number $indexNumber is invalid   
 because it contains an empty string.</p>";

}

}

------------------------------------

ValidateCreditCard.php

else {

$creditCardNumber = $cardNumber;

$creditCardNumber = str\_replace("-", "",   
 $creditCardNumber);

$creditCardNumber = str\_replace(" ", "",   
 $creditCardNumber);

echo "<p>Credit Card Number $indexNumber   
 $creditCardNumber removed dashes and   
 spaces.</p>";

}

------------------------------------

ValidateCreditCard.php

echo "<p>Credit Card Number $indexNumber   
 $creditCardNumber removed dashes and   
 spaces.</p>";

if (!is\_numeric($creditCardNumber)) {

echo "<p>Credit Card Number $indexNumber   
 $creditCardNumber is invalid because it   
 contains a non-numeric character.</p>";

}

else {

echo "<p>Credit Card Number $indexNumber   
 $creditCardNumber is a valid Credit Card   
 Number.</p>";

}

**Part 2**

CompareStrings.php

<!doctype html>

<html>

<head>

<!--

Project 02\_03\_01

Author: Mark J. Buckler

Date: October 3, 2017

Filename: CompareStrings.php

-->

<title>Compare Strings</title>

<meta charset="UTF-8">

<meta name="viewport" content="initial-scale=1.0">

<script src="modernizr.custom.65897.js"></script>

</head>

<body>

<h2>Compare Strings</h2>

<?php

?>

</body>

</html>

------------------------------------

CompareStrings.php

<?php

$firstString = "Geek2Geek";

$secondString = "Geezer2Geek";

?>

CompareStrings.php

if ( !empty($firstString) && !empty($secondString) ) {

echo "No strings empty.";

}

else {

echo "<p>Either the \$firstString variable or the   
 \$secondString variable does not contain a value so   
 the two strings cannot be compared.</p>";

}

------------------------------------

CompareStrings.php

echo "No strings empty.";

if ($firstString == $secondString) {

echo "<p>Both strings are the same.</p>";

}

else {

echo "<p> Both strings have " .   
 similar\_text($firstString, $secondString) . "   
 character(s) in common.</p>";

echo "<p>You must change " .   
 levenshtein($firstString, $secondString) . "   
 character(s) to make the strings the same.</p>";

}

**Part 3**

ValidateLocalAddress.php

<!doctype html>

<html>

<head>

<!--

Exercise 02\_03\_01

Author: Mark J. Buckler

Date: October 3, 2017

Filename: ValidateLocalAddress.php

-->

<title>Validate Local Address</title>

<meta charset="UTF-8">

<meta name="viewport" content="initial-scale=1.0">

<script src="modernizr.custom.65897.js"></script>

</head>

<body>

<h2>Validate Local Address</h2>

<?php

?>

</body>

</html>

------------------------------------

ValidateLocalAddress.php

<?php

$email = array("jsmith123@example.org",

"john.smith.mail@example.org",

"john.smith@example.org",

"john.smith@example",

"jsmith123@mail.example.org");

?>

ValidateLocalAddress.php

foreach ($email as $emailAddress) {

echo "The email address &ldquo;" . $emailAddress .   
 "&rdquo; ";

if (preg\_match("/^(([A-Za-z]+\d+)|" . "([A-Za-z]+\.[A-  
 Za-z]+))" . "@((mail\.)?)example\.org$/i",   
 $emailAddress) == 1) {

echo " is a valid e-mail address.<br>";

}

else {

echo " is not a valid e-mail address.<br>";

}

}