Oumou Balde

BIS 31

Pr Tanweer

Exercice 21

December 20, 2017

Exercice 21: Credit card Validation

In this project, you will create a script that validates whether a credit

card number contains only integers. The script will remove dashes

and spaces from the string. After the dashes and spaces are removed,

the script should reject the credit card number if it contains any other

non-numeric characters.

Create a new document in your text editor.

2. Type the <!DOCTYPE> declaration, <html> element, document

head, and <body> element. Use the strict DTD and “Validate

Credit Card” as the content of the <title> element.

3. Add the following text and elements to the document body:

<h1>Validate Credit Card</h1><hr />

4. Add the following script section to the document body:

<?php

?>

5. Declare a $CreditCard array that contains three values: an

empty string, a valid credit card number with numbers and

dashes, and a credit card number with four initial uppercase

letter Os.

$CreditCard = array(

"",

"8910-1234-5678-6543",

"OOOO-9123-4567-0123");

6. Add the following statements to iterate through each of the

elements in the $CreditCard array to determine if the element

contains a value.

foreach ($CreditCard as $CardNumber) {

if (empty($CardNumber))

echo "<p>This Credit Card Number is

invalid because it contains an empty

string.</p>";

7. Add the following else clause to validate the credit card number.

Th e code uses str\_replace() functions to remove any dashes

and spaces in the number. Th en, a nested if...else statement

checks whether the new value is numeric. If the number is not

numeric, a warning is displayed. If the number is numeric, the

modifi ed credit card number is displayed in the Web browser.

else {

$CreditCardNumber = $CardNumber;

$CreditCardNumber = str\_replace("-", "",

$CreditCardNumber);

$CreditCardNumber = str\_replace(" ", "",

$CreditCardNumber);

if (!is\_numeric($CreditCardNumber))

echo "<p>Credit Card Number " .

$CreditCardNumber . " is not a valid

Credit Card number because it contains

a non-numeric character. </p>";

else

echo "<p>Credit Card Number " .

$CreditCardNumber . " is a valid

Credit Card number.</p>";

}

}

8. Save the document as ValidateCreditCard.php in the

Projects directory for Chapter 3 and upload the fi le to the

server.

9. Open ValidateCreditCard.php in your Web browser by entering

the following URL:

http://<yourserver>/PHP\_Projects/Chapter.03/Projects/

ValidateCreditCard.php. Test the script to see if it displays a

message for an empty string, strips dashes and spaces from

the credit card numbers, and identifi es which credit card

numbers are valid.

10. Close your Web browser window.

**Anwser Code**

<!DOCTYPE>

<html>

<head>

<title>Validate Credit Card</title>

<meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />

</head>

<body>

<h1>Validate Credit Card</h1><hr/>

<?php

$CreditCard = array(

"",

"8910-1234-5678-6543",

"OOOO-9123-4567-0123");

foreach ($CreditCard as $CardNumber) {

if (empty($CardNumber))

echo "<p>This Credit Card Number is invalid because it

contains an empty string.</p>";

else {

$CreditCardNumber = $CardNumber;

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

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

if (!is\_numeric($CreditCardNumber))

echo "<p>Credit Card Number " .

$CreditCardNumber.

"is not a valid Credit".

"Card number because it contains" .

"a non-numeric character. </p>";

else

echo "<p>Credit Card Number " .

$CreditCardNumber .

" is a valid Credit Card number.</p>";

}

}

?>

</body>

</html>

**Result of the code**

