

## **Practical No. 9 – String handling in PHP.**

**Name of Programmer – Vishal Pravin Jatti**

**Roll No. – 26**

**Code:**

```
<?php
$string = "Hello, this is a string example.";

// String length
$length = strlen($string);
echo "Length of the string: $length <br>";

// Convert to lowercase
$lowercase = strtolower($string);
echo "Lowercase: $lowercase <br>";

// Convert to uppercase
$uppercase = strtoupper($string);
echo "Uppercase: $uppercase <br>";

// Substring extraction
$substring = substr($string, 7, 5); // Extracts 'this '
echo "Substring: $substring <br>";

// String search
$search = "string";
if (strpos($string, $search) !== false) {
    echo "The word '$search' is found in the string. <br>";
} else {
    echo "The word '$search' is not found in the string. <br>";
}

// String replace
$replace = str_replace("is", "was", $string);
echo "Replacing 'is' with 'was': $replace <br>";

// String repetition
$repeatedString = str_repeat($string, 3);
echo "Repeated string: $repeatedString <br>";

// Trim leading and trailing whitespace
$whitespaceString = " Trim whitespace ";
$trimmedString = trim($whitespaceString);
echo "Trimmed string: '$trimmedString' <br>";

// Explode string into an array
$phrase = "This is a phrase";
$words = explode(" ", $phrase);
echo "Exploded string into an array: ";
```

```

print_r($words);
echo "<br>";

// Implode array into a string
$implodeString = implode(" ", $words);
echo "Imploded array into a string: $implodeString <br>";

// Reverse a string
$reversedString = strrev($string);
echo "Reversed string: $reversedString <br>";

// Check if string starts/ends with a specific substring
if (substr($string, 0, 5) === "Hello") {
    echo "String starts with 'Hello'. <br>";
}

if (substr($string, -8) === "example.") {
    echo "String ends with 'example.'. <br>";
}

// Character count
$charCount = count_chars($string, 1);
echo "Character count: <br>";
foreach ($charCount as $char => $count) {
    echo "Character '$char' appears $count times. <br>";
}

// HTML Special Characters
$htmlString = '<p>This is a paragraph.</p>';
$encodedString = htmlspecialchars($htmlString);
echo "Encoded HTML entities: $encodedString <br>";

// Compare strings
$string1 = "Hello";
$string2 = "hello";
$comparison = strcmp($string1, $string2);
if ($comparison === 0) {
    echo "The strings are equal. <br>";
} elseif ($comparison > 0) {
    echo "'$string1' is greater than '$string2'. <br>";
} else {
    echo "'$string1' is less than '$string2'. <br>";
}
?>

```

**Output:**

```
localhost/WEBpract/pract9.php x +
localhost/WEBpract/pract9.php

Length of the string: 32
Lowercase: hello, this is a string example.
Uppercase: HELLO, THIS IS A STRING EXAMPLE.
Substring: this
The word 'string' is found in the string.
Replacing 'is' with 'was': Hello, thwas was a string example.
Repeated string: Hello, this is a string example.Hello, this is a string example.Hello, this is a string example.
Trimmed string: 'Trim whitespace'
Exploded string into an array: Array ( [0] => This [1] => is [2] => a [3] => phrase )
Imploded array into a string: This, is, a, phrase
Reversed string: .elpmaxe gnirts a si siht ,olleH
String starts with 'Hello'.
String ends with 'example.'.
Character count:
Character '32' appears 5 times.
Character '44' appears 1 times.
Character '46' appears 1 times.
Character '72' appears 1 times.
Character '97' appears 2 times.
Character '101' appears 3 times.
Character '103' appears 1 times.
Character '104' appears 1 times.
Character '105' appears 3 times.
Character '108' appears 3 times.
Character '109' appears 1 times.
Character '110' appears 1 times.
Character '111' appears 1 times.
Character '112' appears 1 times.
Character '114' appears 1 times.
Character '115' appears 3 times.
Character '116' appears 2 times.
Character '120' appears 1 times.
Encoded HTML entities: <p>This is a paragraph.</p>
'Hello' is less than 'hello'.
```

---