Great Place To Work. Certified MAR 2021-FEB 2022 INDIA

UPES
UNIVERSITY OF THE FUTURE

UPES

UNIVERSITY OF TOMORROW

nirf RANKING 2021

Ranked amongst **top 100** universities in **India**

Accredited **Grade** 'A' by NAAC

QS 5 Star Rating for Academic Development, Employability, Facilities and Program Strength

Perfect score of **150/150** as a testament to exceptional E-Learning methods

University of the Year (North India) awarded by ASSOCHAM

Certified for **safety and hygiene** by Bureau Veritas

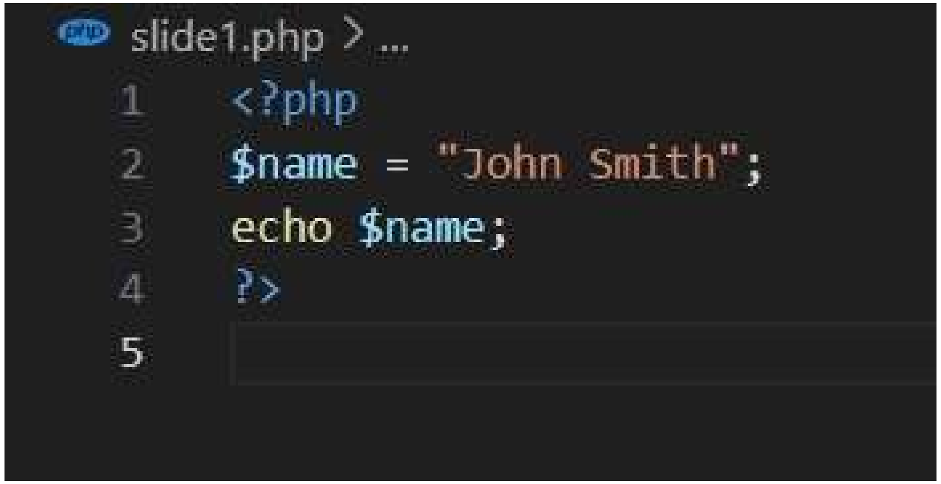
Exploring PHP String and Date & Time Functions

Made By Diganta Datta & Khushi Jain

PHP Strings

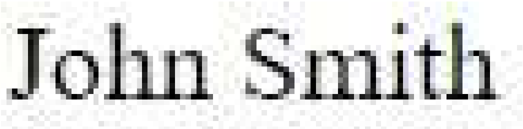
- PHP Strings: Sequences of characters like “Hello World”.
- Declaration and Initialization:
  - Use single (' ') or double (" ") quotes.
  - Example: ‘$name=“John Smith”’

PHP String Functions Example



```php
<?php
$name = "John Smith";
echo $name;
?>
```

OUTPUT:



```
John Smith
```

PHP String Concatenation

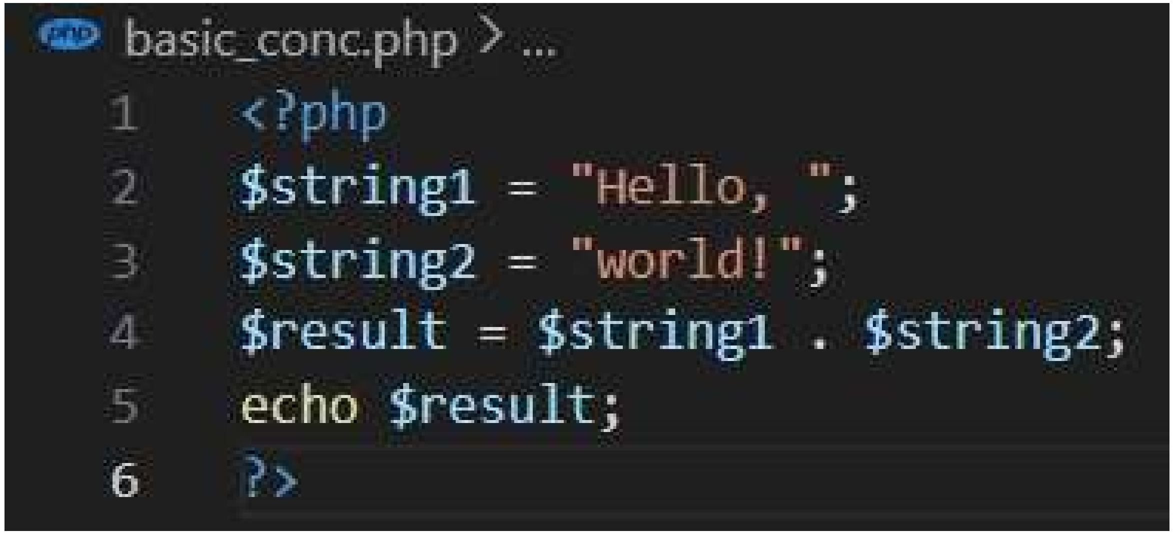
- String concatenation is the process of joining multiple strings together to create a single string.
- In PHP, the primary operator for concatenating strings is the "." (dot) operator.
- Example:

```
echo "Welcome to " . "PHP programming.";
```

PHP String Concatenation Types

1. Basic Concatenation
2. Concatenation in Output
3. Concatenation with Variables
4. Concatenation Assignment Operator (.=)

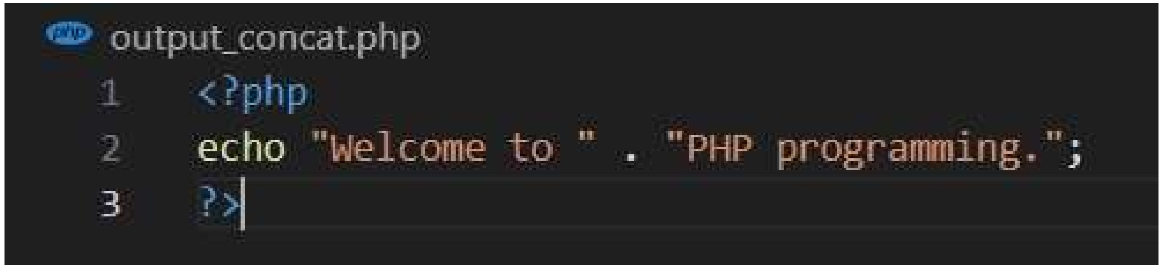
Example of Basic Concatenation in PHP



basic_conc.php > ...

```php
<?php
$string1 = "Hello, ";
$string2 = "world!";
$result = $string1 . $string2;
echo $result;
?>
```

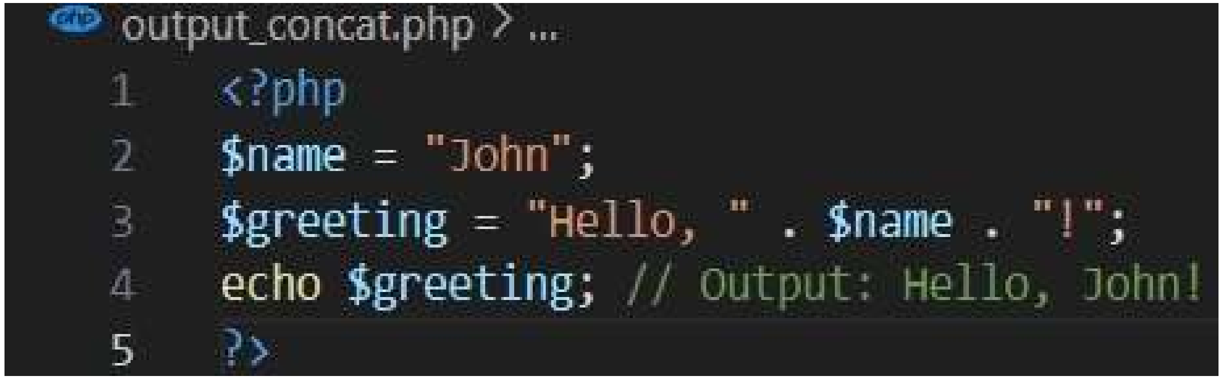
Example of Concatenation in Output in PHP



output_concat.php

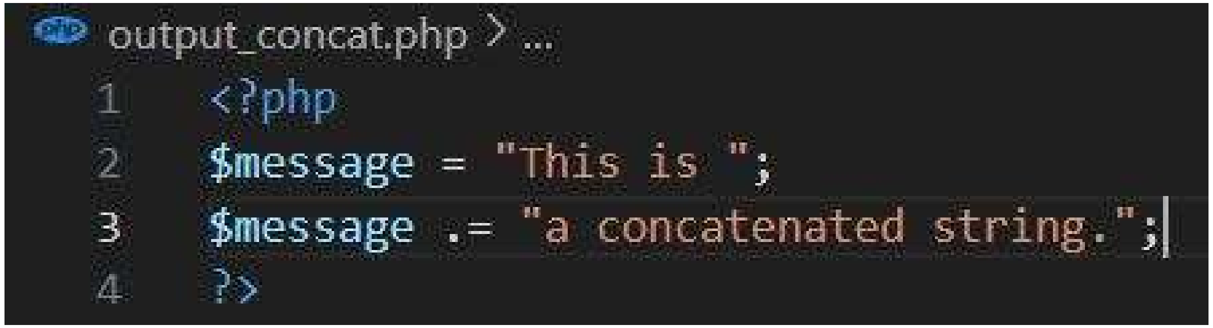
```php
<?php
echo "Welcome to " . "PHP programming.";
?>
```

Example of Concatenation with Variables in PHP



```php
<?php
$name = "John";
$greeting = "Hello, " . $name . "!";
echo $greeting; // Output: Hello, John!
?>
```

Example of Concatenation Assignment Operator (.=)



```php
<?php
$message = "This is ";
$message .= "a concatenated string.";
?>
```

String Functions

- **strlen() - String Length -** function to get the length of a string.

Example:

```
$text = "Hello, World!";
$length = strlen($text);
echo "The length of '$text' is $length characters.";
```

- **strpos() - Finding Substrings -** finding the position of a substring in a string. It helps locate the starting position of the first occurrence of the substring.

Example:

```
$text = "The quick brown fox jumps over the lazy dog.";
$substring = "fox";
$position = strpos($text, $substring);
echo "The position of '$substring' in the text is at index $position.";
```

String Functions

- **substr() - Extracting Substrings -** extracting parts of a string.

Example:

```
$text = "The quick brown fox";
$substring = substr($text, 10, 5);
echo "Substring: $substring";
```

- **str_replace() - Replacing Text -** function for text replacement.

Example:

```
$text = "Hello, World!";
$newText = str_replace("World", "Universe", $text);
echo "Updated Text: $newText";
```

String Functions

Some other Important Functions:

- Strtolower()
- Strtoupper()
- Ucfirst()
- Strrev()
- Char()
- Ord()
- Htmlspecialchars()

REFER W3SCHOOL LINK For Further Studies………

ACTIVITY FUN TIME 1 (PHP STRING)

TASK 1:
FIND LENGTH OF THE STRING “I am currently enrolled in the MCA program at UPES”.

TASK 2:
CONCATINATE “ ‘your city name as coder’s input’ ” at last of the string before.

TASK 3:
REPLACE your city name with “Dehradun”

*(**RULES**: Make a team of 2, and also show the final result to get the marks)*

PHP Date & Time Functions

Introduction to Date & Time in PHP:

- PHP provides robust tools for working with date and time.
- Handling date and time is crucial in many web applications for tasks like event scheduling, logging, and data presentation.

Importance in Web Applications:

- Explain the significance of date and time in web development:
  - Displaying dynamic content with timestamps.
  - Tracking user activity and interactions.
  - Handling time-sensitive operations, such as expiring tokens or sessions.
  - Managing schedules and appointments in various applications.

Built-in Date & Time Functions:

- **time() -** Function is a built-in function that is used to retrieve the current Computer timestamp. It is commonly used to capture the current date and time in PHP scripts. This timestamp is useful for various purposes, including date and time calculations, logging, and tracking events.

Example:

```
$currentTimestamp = time();
echo "Current Unix timestamp: $currentTimestamp";
```

Built-in Date & Time Functions:

- **strtotime() -** PHP function used to parse a human-readable date string and convert it into a Computer timestamp. It's commonly used when you have a date in a user-friendly format (e.g., "January 1, 2023" or "next week") and you want to convert it into a timestamp for various date and time calculations.

Example:

```
$dateString = "January 1, 2023";
$timestamp = strtotime($dateString);
echo "Unix timestamp for $dateString: $timestamp";
```

Built-in Date & Time Functions:

- **strtotime() -** PHP function that creates a new DateTime object, which represents a specific date and time. It is used to create DateTime objects for various date and time operations, such as formatting, comparison, addition, and subtraction.

Example:

```
// Create a DateTime object for the current date and time
$now = date_create();

// Create a DateTime object for a specific date and time
$date = date_create("2023-12-31 23:59:59");

// Create a DateTime object with a specific time zone
$timezone = new DateTimeZone("America/New_York");
$dateInNY = date_create("2023-12-31 23:59:59", $timezone);
```

Built-in Date & Time Functions:

- **date_diff() -** Calculates the difference between two DateTime objects and returns a DateInterval object representing the interval. It is commonly used when you need to measure the time elapsed between two events or calculate the age difference between two dates.

Example:

```
$startDate = date_create("2022-01-01");
$endDate = date_create("2022-12-31");
$interval = date_diff($startDate, $endDate);
echo "Difference: " . $interval->format("%y years, %m months, %d days");
```

Built-in Date & Time Functions:

- **mktime() -** Creates a Computer timestamp based on the specified date and time components and returns it as an integer. It is commonly used when you need to generate timestamps for a particular date and time, especially when dealing with date-related calculations.

Syntax:

mktime(int $hour, int $minute, int $second, int $month, int $day, int $year, int $is_dst = -1)

Example:

```
$timestamp = mktime(12, 0, 0, 3, 15, 2023);
echo "Unix timestamp for March 15, 2023, 12:00 PM: $timestamp";
```

ACTIVITY FUN TIME 2 (PHP DateTime Function)

SHOW IN PHP OUTPUT IN USING DATETIME FUNCTIONS

TASK 1:
On what date did Neil Armstrong set foot on the moon during the Apollo 11 mission?

TASK 2:
The next occurrence of a solar eclipse visible from India will be on April 8, 2024. Calculate how many days are left until that eclipse.

TASK 3:
If it's 12:00 PM (noon) in London, what time is it in DELHI?

TASK 4:
What will be the day exactly one year from today?

*(**RULES**: Make a team of 2, and also show the final result to get the marks)*

THANK YOU

UPES

UNIVERSITY OF THE FUTURE