

Introduction to PHP

Chencha Jacob

Sample PHP script

```
<?php
$today=date("l");
$topics=array(
    "Monday"=>"php",
    "Tuesday"=>"html",
    "Wednesday"=>"css",
    "Thursday"=>"apache",
    "Friday"=>"markdown"
);

$topic= $topics[$today];
echo "Today $today we are going to learn all about $topic";
```

Anatomy of a PHP application

```
<?php //All php scripts must start with this line. It acts as a marker for the interpreter
$today=date("l"); //PHP internal function
$topics=array("Monday"=>"php",
    "Tuesday"=>"html",
    "Wednesday"=>"css",
    "Thursday"=>"apache",
    "Friday"=>"markdown"
); //An array, special type of php variable
$topic= $topics[$today];//Accessing an array, assignment array
echo "Today $today we are going to learn all about $topic"; //Prints out to screen
```

Try it yourself

- Try changing the code
 - Try a new parameter in the `*date*` function
 - Change a lesson
 - Print out your name instead of `*we*`
- Try breaking the code
 - Remove the PHP tags
 - Remove Colons
 - Accessing unavailable values

Starting local server

Run

```
php -S localhost:8888 #Starts listening for requests on port 8888
```

You can access your script from your browser *localhost:8000/{filename}*

Variables

We have already seen some variables:

- *\$today*
- *\$topics*
- *\$topic*

Variables are containers for data. They can represent different values. You generally use them to store information slotted for retrieval later on.

```
$name= "Chencha";  
echo $name; //Prints out *Chencha*  
$name= "Lucy"  
echo $name //Prints out *Lucy*
```

Properties of a PHP variable include:

- A variable starts with the \$ sign, followed by the name of the variable
- A variable name must start with a letter or the underscore character
- A variable name cannot start with a number
- A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _)
- Variable names are case-sensitive (*ageand*AGE are two different variables)

Functions

Just like variables we have also seen functions before in this case *date*

A function can be thought of a recipe that gives you useful results when called. For example we use the *date* function to give us today's day.

Functions must have a parenthesis even if they have no arguments.

You can modify behaviour of a program by providing arguments. NEVER use globals.

Usually the inbuilt function set defines the API of the language.

Some useful properties of functions include:

- A function is a block of statements that can be used repeatedly in a program.
- A function will not execute immediately when a page loads.
- A function will be executed by a call to the function.

When stuck

- Try google, 99% of your problems can be sorted this way.
- Look at the official documentation. [php.net](#)
- Look at [w3schools](#)

Task

Task

Handling names is a common task in web applications. However names can be given without standard structure, ie

- cHENCHA jACOB
- ChenchA Jacob
- ChenchA jacob

The task required is to store your name in a variable and print it out to the screen in capital. That is for the examples above, your script would print out *CHENCH A JACOB*

Assignment

Write a function that takes in a *string* as an argument and returns the *string* in sentence case.

eg Given “chencia jacob” it gives “Chencia Jacob”