Introduction to PHP

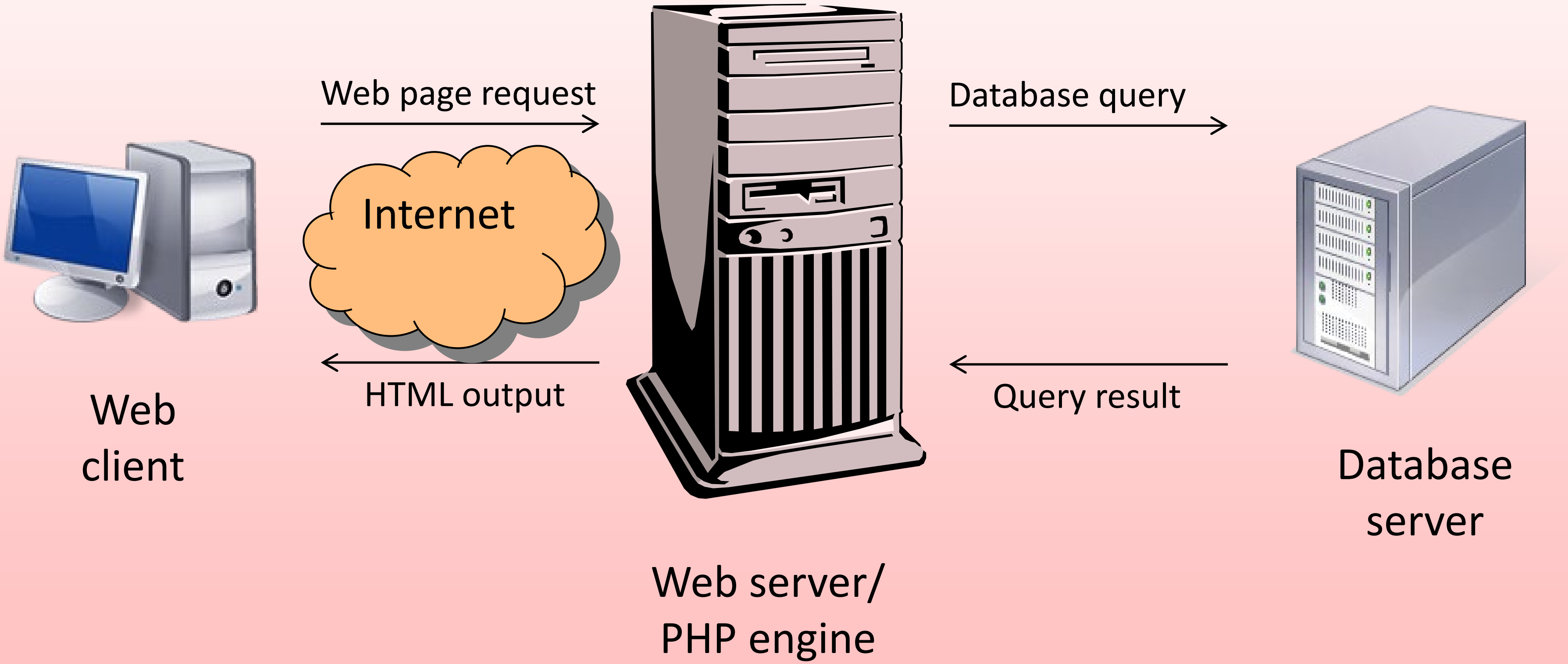
Thomas Li

What is PHP?

- PHP is a server-side technology that builds web pages dynamically.
- e.g. Instead of updating your web pages of an online catalog each time when a product is added or removed, you can store the product details in a database and use PHP to query the database and build the catalog page dynamically.

How does PHP work?

3-tier Architecture



How does PHP work?

When a browser requests a web page that uses PHP:

- The web server sends the page to the PHP engine (residing on the server) for processing.
- If the PHP script doesn’t need to communicate with the database, the PHP engine generates the HTML output, and the web server sends it back to the browser.
- If the script needs to interact with the database, the PHP engine communicates with the database server. When the results come back, the PHP engine puts everything together, and then the web server sends the resulting web page back to the browser.

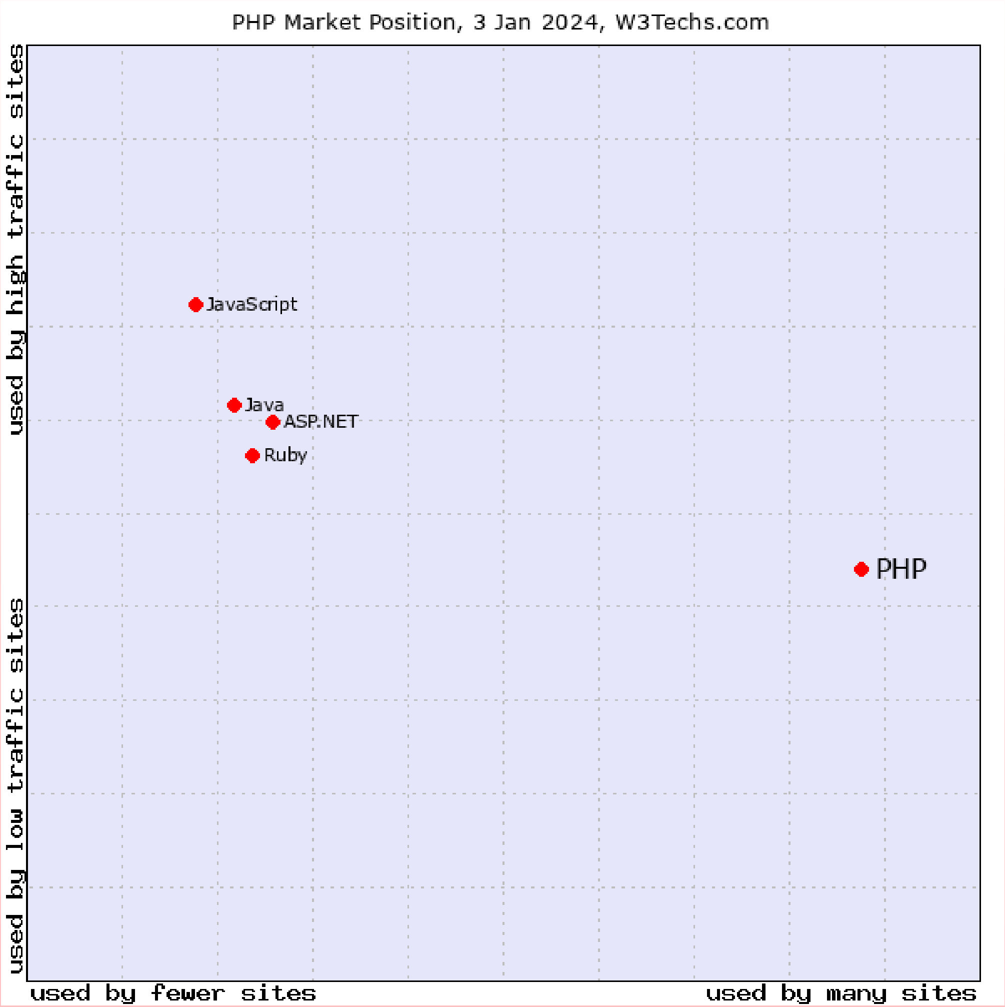
Why PHP?

PHP has the following advantages:

- PHP runs on Windows, Mac OS X, and Linux. With only a few minor exceptions, code written on one operating system works on any of the others (cross-platform).
- It’s open source and free.
- It’s widely available.
- It’s relatively easy to learn.
- There’s a large community of active users, so help is easy to find.
- It’s simple to incorporate into a small website, yet powerful enough to drive some of the busiest websites, including Facebook, Wikipedia, and Yahoo!

Why PHP?

- A survey (by w3techs.com, 2024 found that PHP is used by 76.7% of all websites.
- Many skills you learn about PHP are transferable to other server-side technologies.



Why MySQL

- PHP is often used in conjunction with MySQL.
- MySQL is the most popular open-source database.
- Fast, powerful and well suited for use in websites.
- MySQL works on all major operating systems, so you can develop on one system and later transfer your database to another.
- Used by high-traffic websites like Flickr, Facebook, and YouTube

Our Development Environment

- Since PHP code needs to be processed by a web server, you need a web server set up
- Elements you need:
  - A web server (Apache or Microsoft IIS)
  - PHP engine
  - A database server (MySQL)
- Use an all-in-one package for Windows environment:

WampServer: www.wampserver.com/en

or MAMP for MacOS.

Setup in Adobe Dreamweaver

- Dreamweaver functions like an IDE.
- It provides the functions to:
  - Edit the files
  - Syntax coloring
  - Autocompletion of defined variables
  - Upload the files to server
  - Live view of the pages
  - etc.

Embedding PHP Code in a Page

- The web server needs to recognize which page contains PHP code, and which part in a page is HTML code, which part is PHP code.
  - Name your pages with a ***.php*** filename extension (which may be hidden in your computer)
  - You signal the start of any PHP code within a page by using the opening tag: `<?php`
  - You signal the end of a PHP code block with a closing tag: `?>`
  - You can have as many PHP code blocks within a page as you like
  - PHP code blocks can present anywhere within a page.

Some Basics of PHP

- PHP codes are interpreted by the PHP engine in real time when the code is executed. So PHP is known as a ***scripting*** language.
  - c.f. You need to compile a Java program before you run it, but you don't need to compile a PHP page before you load it.
- Because PHP scripts can ***ONLY*** be executed ***ON A WEB SERVER***, to see the result of your PHP page, you must open the page via a URL in the browser that begins with **`http://`**

How PHP Makes Pages Dynamic

PHP uses a number of techniques common to most programming languages:

- ***Variables***: store information that isn’t fixed or known in advance
- ***Arrays***: store multiple values, usually of related information
- ***Conditional statements***: make decisions, based on some conditions
- ***Functions***: perform tasks in bulk, usually repeatedly
- ***Operators***: add, subtract, multiply, divide, …
- ***Loops***: perform repetitive actions

Using Variables

- What makes programming languages so powerful is their ability to handle unknown values.
- ***Variables*** are containers used to temporarily store values.
  - A PHP variable can store one of the following types of data:

Type	Description
Boolean	True or false (same as Java)
Float	A floating point number (same as double in Java)
Integer	A whole number (same as int in Java)
String	Text (same as String class in Java)
Array	An ordered collection of values
Object	A sophisticated data type that can store and manipulate values
Resource	A reference to an external resource, such as a database result or file
NULL	A variable with no value

Using Variables

- PHP is known as a ***weakly typed language***.
  - You need not specify what type of data a variable will be used for.
  - c.f. Java is known as a ***strongly typed language***.
- All PHP variables must follow the following rules:
  - Variable's name must start with a dollar sign ($)
  - Variable's name can contain a combination of letters, numbers, and underscore. e.g. `$my_report1`
  - The first character after the dollar sign must be either a letter or an underscore. (cannot be a number)
  - Variable names are case-sensitive. i.e. `$name` and `$Name` are different variables.

Using Strings

- You can use either single (') or double (") quotation marks to encapsulate the characters.
  - e.g. :

    ```
    $first_name = "Thomas";
    $today = '12 January, 2023';
    ```

- If the string contains quotation marks, you can use the other quotation marks or escape characters (\)
  - e.g. :

    ```
    $var = 'Define "electronics", please.';
    $var = "Define \"electronics\", please.";
    ```

Sending Data to Web Browser

- To create dynamic web page, all or part of the page content (in HTML format) must be generated by the server-side script (i.e. PHP code)
- **`echo`** and **`print`** are two common functions of PHP for sending the output of a PHP script to a HTML file.

Using Strings

- To print value of a string variable

```
echo $first_name;
```

- To print value of a string variable within a context

```
echo "Hello, $first_name";
```

Writing Comments

- In HTML you can add comments using special tags:

  ```
  <!-- This is a comment. -->
  ```

  - HTML comments are viewable in the source but do not appear in the rendered page.
- In PHP you can add comments in three ways:
  - Everything following two forward slashes (`//`)is ignored until the end of the line
  - Everything following a pound sign (`#`) is ignored until the end of the line
  - Everything between `/*` and `*/` is treated as a comment. This type of comment can stretch across multiple lines.

Concatenating Strings

- Concatenating strings is appending strings together to form a longer string.

```
$full_name = $first_name." ".$last_name;
```

Single Quotation vs Double Quotation

- In representing strings, the use of single quotation mark and double quotation mark is quite similar.
- But note the following difference:
  - Values enclosed within single quotation marks will be treated literally.
  - Values enclosed within double quotation marks will be interpreted.

Introducing Numbers

- Numbers include integers and floating-point numbers
  - e.g. 8, 3.14, 9.83787452, -49.34

Operator	Description
+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Modulus
++	Increment
--	Decrement