

[Home](#)[Knowledgebase](#)[Thoughts](#)[Quotes](#)[About](#)[Contact](#)

Kamalika's Notebook

thinking nothing.....!!

You are here: [Home](#) / [Technology](#) / [PHP](#) / PHP vs. Java which is a better Object Oriented Language?

PHP vs. Java which is a better Object Oriented Language?

February 9, 2015 By [Kamalika](#)

PHP vs. Java

Java was originally developed at Sun Microsystems around 1995. Java applications are compiled to an intermediate byte code, which is run by a virtual machine (JVM). PHP was born in the same year, while Java was meant for client applications and in-browser applets; PHP was specifically created for web as a server side scripting language to embed in HTML pages.

PHP and Java have a vast array of differences, though PHP has borrowed its Object Oriented paradigm from Java even when it comes to OOP using PHP the differences are significant.

Typing

Java is strongly typed PHP is not. Each and every variable in Java must have a declared type, while Java is built over static typing PHP uses dynamic typing; in PHP variables

assume the type of value contained in them and they can change the type implicitly to support type-casting and conversions. This makes a huge difference to method signatures; you can only force method parameters to be of a certain class or interface or an array but you cannot type hint for primitives. So, ***public function myMethod (int \$foo, boolean \$bar){}*** is invalid and will throw a parse error.

Return Type

PHP does not require (or even support) specifying the return type of a function.

Execution Model

PHP classes, functions and data structures, when they do not leverage external infrastructure like cache or database servers, are created in a script and they are garbage-collected at the end of the request. Java applications are instead kept in memory between requests, and the architecture of these two kinds of applications is fundamentally different from each other. Though, one execution model cannot be considered superior to the other. PHP pulls in execution only what it needs, and it pays back with the inability to run periodical tasks such as cron jobs. Java applications can start multiple threads, but their management is much more complex, from the compiling phase to the deployment which includes servlets reloading.

Infrastructure

PHP is easy and simple to deploy in its core form i.e., .php scripts, but using core PHP is not sufficient for the developer to build standard infrastructure and there has to be some framework, which builds the same features over the simple PHP interpreter. Ironically, these framework are similar to Java ones; for example Zend Framework's controllers are the equivalent of servlets: classes with a standard constructor that extend a common base class and act on a request object to produce a response one.

Java has less native features built in the language, as it is not strictly oriented to the web, but it has them in frameworks, which adhere to a standard, the servlet containers. PHP capabilities are hindered by the absence of standards.

For instance, the web.xml file in WAR packages, which represent a web application, define routes to map URLs to servlets.



Tweet



Pages: [1](#) [2](#) [3](#)

Filed Under: [PHP](#)

Tagged With: [Java OOP](#), [Java versus PHP](#), [Object Oriented Programming](#), [PHP OOP](#)

« [How to Find Only Unread Messages in Gmail](#)

[PHP and MVC Design Pattern](#) »

Connect with me online



Follow my notebook

Enter your email address to follow my notebook and receive notifications of new posts by email.

Name *

Email *

Follow

A Little About Me



Technology
Freak. Educator.
Mentor. Digital
Marketing
Enthusiast.
Amateur

Photographer. Avid Reader. Music
Lover. Movie Buff. Foodie. Lover of
WordPress. Ex Disney Employee. For
more, follow me at @kamalika. [Read
More...](#)

Looking for something?

Copyright © 2014 · Kamalika Guha Roy