

PHP vs Javascript

Which sucks harder?



Syntax

- Even easier to learn!
 - C/C++-Syntax
 - The best of Perl (PCRE, Arrays, Environment)
- `mysql_connect()` or `die(mysql_error());`
- Prevents you from writing too complicated code
 - PHP 4: `$my_obj->method()->method();`
 - PHP 5: `$my_array['key']->method();`
 - Something like

```
[2, 3, 5].each(function (value) { value.times(  
function () { alert('hello world!'); } ) } );
```


is overdesigned crap and unreadable!

Core Data Types

- A lot of cool data types
 - Scalar: boolean, integer, float, string
 - Compound: array, object
 - Special: resource, NULL
- Dynamically typed
 - "If it walks like a duck and quacks like a duck, it must be a duck."
 - Broke the ice for all dynamically typed languages
- Unicode as of PHP6
 - Before with mbstring (iconv, utf_*)

Scope / Name spacing / Packaging



- Scope
 - Classes and functions as name spaces (instead of functions as classes, Mister Javascript!): KISS
 - Ever messed around with .call, .apply, .bind in JS?
- Namespacing
 - Real Programmers keep track where their variables come from (no time lost with namespacing)
- Packaging
 - A lot of prefixing standards to choose from!
 - `Zend_Service_Yahoo_ImageResultSet`
 - `HTML_QuickForm_Action_Display`

Standard Library

- In PHP land, **YOU** decide what libs you need
 - Comes with a ton of libs by default
 - Easily configurable in your php.ini
- 3079 well documented standard functions
 - Want to convert Gregorian date to a Julian daycount? `GregorianCalendarToJD()`!
 - Need to sort? No Problem!
 - `sort`, `arsort`, `asort`, `krsort`, `ksort`,
`natsort`, `natcasesort`, `rsort`, `usort`,
`array_multisort`, `uasort`, `uksort`, `dbx_sort`,
`imap_sort`, `ldap_sort`, `yaz_sort`

Extension Libraries

- More than 150 extensions
- PECL, PEAR
- eZ components, Zend Framework and the likes
- About 424'612 frameworks, choose the one that suits you best!
- Java connectivity

Procedural Programming

- All the control statements you know and love
 - if, else, for, while, do...while, switch
- A lot of functions (still 3079)!
- No OOP crap
- Ever tried to explain JS stuff to beginners?
 - sleep() in Javascript?
 - Doing *anything* without objects?

Object Oriented Programming

- Real OOP, Java-like
 - Inheritance
 - Simple, no weird prototyping
 - No multiple inheritance to prevent confusion!
 - Constructors / Destructors
 - Visibility
 - Magic
 - setters, getters, callers
 - __toString(), __sleep(), __wakeup(), __clone(), ...
 - No messing with scope

Functional Programming

- Not needed in the real world of programming
- Can be done:
 - `create_function()`: Create lambda-style functions
 - Metaphp: "Take over the world"
- Just as cool as Javascript!

Meta Programming



- You see it everywhere in PHP!

- Code generation
 - Parser-Generation
 - Unit tests
 - Cache files
- Templating
- Database abstraction
- SPL (ArrayAccess, Iterator)
- Stream wrappers
- Magic setters, getters, callers
- Reflection (see you later)

Aspect-Oriented Programming

- Extensions / Modifications
 - PHPAspect
 - Aspect-Oriented PHP
 - aspectPHP
- Roll your own
 - AOP Library for PHP

Networking

- Better than Javascript's
 - Can communicate with anything over anything
 - Sockets: <http://www.php.net/sockets>
 - CURL: <http://www.php.net/curl>
 - HTTP: <http://www.php.net/http>
 - SSH2: <http://www.php.net/ssh2>
 - etc.

Error Handling



- Types
 - Standard errors
 - Exceptions
- Handling
 - Custom error and exception handling
 - Full control over if and how user sees error
- Architecture
 - Developer knows about the errors!

Multitasking



- Great way to shoot yourself in the foot
 - PHP is made to be simple
 - No need, PHP doesn't affect client-side UI
 - PHP is about fast content / application delivery
- Still possible
 - Process Control Functions (PCNTL)
 - Command line PHP (CLI)
 - AJAX

Reflection

- Better than Javascript's
 - Reflection API
 - "Ye olde reflection"
 - `get_class()`, `class_get_methods()`, etc.
 - `__CLASS__`, `__LINE__`, etc.
 - `var_dump()`
 - Talk to the engine
 - Tokenizer: <http://www.php.net/tokenizer>
 - Parsekit <http://www.php.net/parsekit>
 - Runkit: <http://www.php.net/runkit>

Garbage Collection

- KISS: reference counting
- In (max) 30 seconds it's all trash!
- When did you last see PHP gobble all your memory?

Security



- Common PHP attack vectors well known
- Auditing tools
 - Spike PHP Security Audit Tool
 - Chorizo: <https://chorizo-scanner.com/>
- PHP Security as University research topic:
 - http://www.seclab.tuwien.ac.at/papers/pixy_techreport.pdf
 - <http://www.seclab.tuwien.ac.at/papers/pixy2.pdf>
 - <http://glide.stanford.edu/yichen/research/sec.pdf>
 - <http://wwwconf.ecs.soton.ac.uk/archive/00000544/01/p40-huang.pdf>

Tools

- Debugging
 - XDebug
 - Zend Debugger / Profiler
 - etc.
- IDEs
 - Zend Studio
 - PHPEclipse
 - Weaverslave
 - Your favorite editor (straightforward syntax)

Server Side

- King of the server-side scripting languages
- Proven architecture with Apache (and IIS)
- Rock-solid **execution model**
- Excellent database support
- PHP CLI as popular scripting language

Client Side

- Great capabilities too
 - PHP GTK: <http://gtk.php.net> (GTK2)
 - PHP-QT: <http://www.php-qt.org/> (QT)
 - Winbinder: <http://winbinder.org/> (MFC)
 - Klorofil: <http://www.klorofil.org/> (x-platoform inc. browser)
 - Curses: <http://www.php.net/ncurses> (shell GUIs)
 - Ming: <http://www.php.net/ming> (Flash)

Final Rave

- PHP is the better Javascript
 - Simpler
 - Faster
 - Better
- A lot more popular than JS on the [list](#) of the most popular programming languages
- Kicks your programming language's ass!