

DESARROLLO EN ENTORNO CLIENTE/SERVIDOR

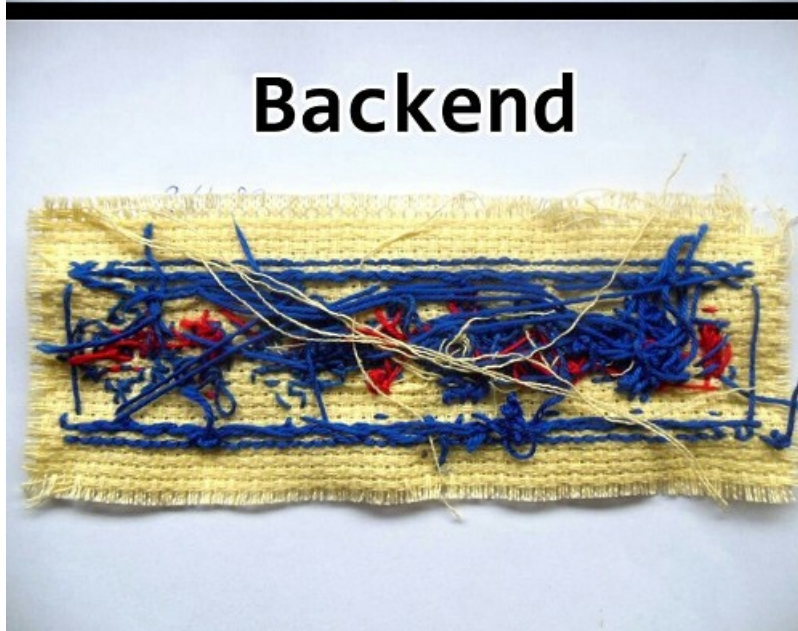
UD2 – FUNDAMENTOS DE PROGRAMACIÓN WEB

Tipos de datos y operadores

Frontend



Backend



UD2 - Fundamentos de programación web

1. TIPOS DE DATOS.....	3
2. OPERADORES.....	5
2.1. Actividad.....	6

1. TIPOS DE DATOS

Las variables almacenan datos, pero no todos los datos son de la misma naturaleza, ya que pueden ser números, cadenas de texto o estructuras complejas, y a esto es a lo que denominamos *tipos de datos*, a qué tipo de información pueden almacenar las variables. Aunque los tipos de datos son similares en todos los lenguajes, existen ciertas particularidades que deben ser tenidas en cuenta a la hora de programar.

Resumen de tipos de datos:

JavaScript	PHP	Python
<ul style="list-style-type: none">• boolean• number• BigInt• string• object• null• undefined• symbol	<ul style="list-style-type: none">• Integer• float• string• boolean• array• iterable• object• callable• resource• null	<ul style="list-style-type: none">• bool• int• float• complex• str• object• Otros tipos: listas, tuplas, conjuntos, diccionarios...

Referencias:

JavaScript: https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Grammar_and_Types

PHP: <https://www.php.net/manual/es/language.types.intro.php>

Python: <https://docs.python.org/es/3/reference/datamodel.html#the-standard-type-hierarchy>

En programación existe el tipado estático o dinámico y el tipado fuerte o débil.

- **Tipado estático:** La comprobación del tipo de la variable se hace en tiempo de compilación (hay que declarar el tipo de las variables).
- **Tipado dinámico:** La comprobación del tipo se hace en tiempo de ejecución (una misma variable puede tener varios tipos).
- **Tipado fuerte:** Una variable de un tipo no se puede usar como si fuese de otro tipo sin una conversión de tipo previa.
- **Tipado débil:** No tiene la restricción anterior, el sistema intenta hacer conversiones de tipo

automáticas. Otorga más flexibilidad, pero también puede generar problemas

En el siguiente esquema se puede ver una clasificación de lenguajes según estas características:

TIPADO	DEBIL	FUERTE
DINAMICO	JavaScript PHP Visual Basic Perl	Python Ruby Groovy
ESTÁTICO	C C++	C# Java Haskell

2. OPERADORES

Los operadores tratan con las variables y realizan con ellos una operación. En función del número de operandos con el que trabajen podemos encontrarnos con operadores unarios (un operando), binarios (dos operandos, los más comunes) y ternarios (tres operandos).

OPERADORES	JavaScript	PHP	Python
Aritméticos:			
Suma	+	+	+
Resta	-	-	-
Multiplicación	*	*	*
Exponente	** (ES2016)	**	**
División	/	/	/
Módulo	%	%	%
Incremento	++	++	++
Decremento	--	--	--
Floor			//
Comparación:			
Igualdad	==	==	==
Igualdad estricta	===	===	
Distinto	!=	!= <>	!=
Distinto estricto	!==	!==	
Mayor	>	>	>
Menor	<	<	<
Mayor o igual	>=	>=	>=
Menor o igual	<=	<=	<=
Nave espacial		<=>	
Lógicos:			
And	&&	&& and	and
Or		or	or
Not	!	!	not
Xor		xor	
Operadores de bit:			
And a nivel bit	&	&	&
Or a nivel bit			
Not a nivel bit	~	~	~
Xor a nivel bit	^	^	^
Desplazamiento izq.	<<	<<	<<

Desplazamiento der.	>>	>>	>>
Desplazamiento der. 0	>>>		
Asignación:			
Igual	=	=	=
Más igual	+=	+=	+=
Menos igual	-=	-=	-=
Multiplicación igual	*=	*=	*=
División igual	/=	/=	/=
Resto igual	%=	%=	%=
Floor igual			//=
Exponente igual	**=	**=	**=
And bit igual	&=		&=
Or bit igual	=		=
Xor bit igual	^=		^=
Desplaz. izq. igual	<<=		<<=
Desplaz. der. igual	>>=		>>=
Cadena igual		.=	
Asignación desestructurada	[]=[]		
Otros operadores:			
Concatenación	+	.	+
Operador ternario	?:	?:	if else (,) []
Fusión de null (null coalescing)	??	??	
Member acces	.	->	.
Pertenencia	in		in
No pertenencia			not in
Identidad			is
No identidad			is not
Tipo	typeof		
Instancia	instanceof		

2.1. Actividad

Trabajando por equipos, vais a realizar un trabajo sobre los operadores de cada uno de los lenguajes. A cada grupo se le asignará una familia (o familias de operadores) y ha de entregar un documento que exponga cómo funcionan cada uno de los operadores de la familia asignada. Por cada operador se habrá de indicar (aspectos obligatorios):

- Una descripción de su función y comparación entre JavaScript, PHP y Python.
- Sobre qué tipos de datos funciona (enteros, strings, etc.) y qué comportamiento tiene sobre esos grupos de datos. Este punto es especialmente importante, porque define el alcance de cada operador.
- Precedencia de los operadores, dentro de la familia asignada y en relación a todas las familias restantes.
- Qué límites de operación tiene (posibles errores).
- Al menos un ejemplo de uso (en los casos en que tenga varios usos posibles, un ejemplo por caso), aunque se recomiendan varios ejemplos para mostrar comportamientos singulares, errores, etc.
- Enlaces y bibliografía usada para hacer el trabajo, de forma que sirva como método de consulta para ampliar información.

Al finalizar el trabajo, cada grupo hará una breve presentación al resto de la clase sobre los operadores que ha trabajado y el documento que se entregará se publicará en Aules para el resto de compañeros a modo de apuntes sobre los operadores.

NOTA: Se valorará de forma adicional si se exponen casos reales donde se muestre la aplicación de los operadores tratados, problemas que puedan causar un mal uso o errores por los tipos de datos implicados.