

# Conceptos básicos de PHP (I)

Ricardo Pérez López

IES Doñana, curso 2018-19

- 1 **Introducción a PHP**
- 2 **Sintaxis básica**
- 3 **Funcionamiento del intérprete**
- 4 **Intérprete interactivo**
- 5 **Variables**
- 6 **Tipos básicos de datos**
- 7 **Manipulación de datos**
- 8 **Constantes**
- 9 **Flujo de control**
- 10 **Funciones predefinidas destacadas**

## Sección 1

# **Introducción a PHP**

# Página web de PHP



# Instalación de PHP

```
// Useless comment.  
alert('hi');
```

When this fragment is shown, the first line of code (`span.line`) will have the `"focus"` class added to it.

Another fragment. This time, both lines will now have the `"focus"` class.

# Documentación y búsqueda de información

## Sección 2

### **Sintaxis básica**



# Datos e instrucciones

# Sentencias y comandos

Comando `echo`.

# Expresiones, operadores y funciones

Ejemplos: aritmética, `cos()`, `max()`.

## Sección 3

# Funcionamiento del intérprete

# Modo dual de operación

Se llaman *modo HTML* y *modo PHP*.

# Etiquetas `<?php` y `?>`

## Sección 4

### **Intérprete interactivo**

```
php -a
```





## Sección 5

# **Variables**

# Conceptos básicos

# Destrucción de variables

# Operadores de asignación

- Por valor
- Por referencia

# Variables predefinidas

## Sección 6

# Tipos básicos de datos

# Lógicos

- Operadores lógicos



# Numéricos

- Enteros (`int`)
- Números en coma flotante (`float`)
- Operadores
  - Aritméticos
  - De incremento/decremento

# Cadenas

- Operadores de cadenas
  - Concatenación
  - Acceso y modificación por caracteres
  - De incremento/decremento
- Funciones de manejo de cadenas
- Extensión *mbstring*

# Nulo

## Sección 7

# Manipulación de datos

# Precedencia de operadores

# Operadores de asignación compuesta

# Comprobaciones

- De tipos
  - `gettype()`
  - `is_*()` (poco útiles en formularios, ya que sólo se reciben `strings`)
- De valores
  - `is_numeric()`
  - `ctype_*()`

# Conversiones de tipos

- Conversión explícita (forzado o *casting*) vs. automática
  - A `bool`
  - A `int`
  - A `float`
  - De `string` a número
  - A `string`
- Funciones de obtención de valores (Hacen más o menos lo mismo que los *casting* pero con funciones en lugar de con operadores. Puede ser interesante porque las funciones se pueden guardar, usar con *map*, *reduce*, etc.)
  - `intval()`
  - `floatval()`
  - `strval()`
  - `boolval()`
- Funciones de formateado numérico
  - `number_format()`
  - `money_format()`
    - `set_locale()`



# Comparaciones

- Operadores de comparación
- `==` vs. `===`
- Operador ternario (`? :`)
- Operador de coalescencia o fusión de null (`??`)
- Reglas de comparación de tipos

## Sección 8

### **Constantes**

# Constantes

`define()` y `const`

# Constantes predefinidas

```
defined()
```

## Sección 9

# Flujo de control

# Estructuras de control



# Inclusión de scripts

## Sección 10

### **Funciones predefinidas destacadas**

```
isset()
```

```
empty()
```

```
var_dump()
```

## Sección 11

# Arrays

# Arrays

# Operadores para arrays



# Funciones de manejo de arrays

# foreach

# Conversión a array

## Ejemplo: `$argv` en CLI

## Sección 12

### **Funciones definidas por el usuario**

# Funciones definidas por el usuario

# Argumentos

- Paso de argumentos por valor y por referencia
- Argumentos por defecto

# Ámbito de variables

- Ámbito simple al archivo
- Variables locales
- Uso de `global`
- Variables superglobales



# Declaraciones de tipos

- Declaraciones de tipos de argumento
- Declaraciones de tipo de devolución
- Tipos *nullable* y `void`
- Tipificación estricta

## Sección 13

### **Comentarios y documentación del código**

## Sección 14

### **Ejercicios**