



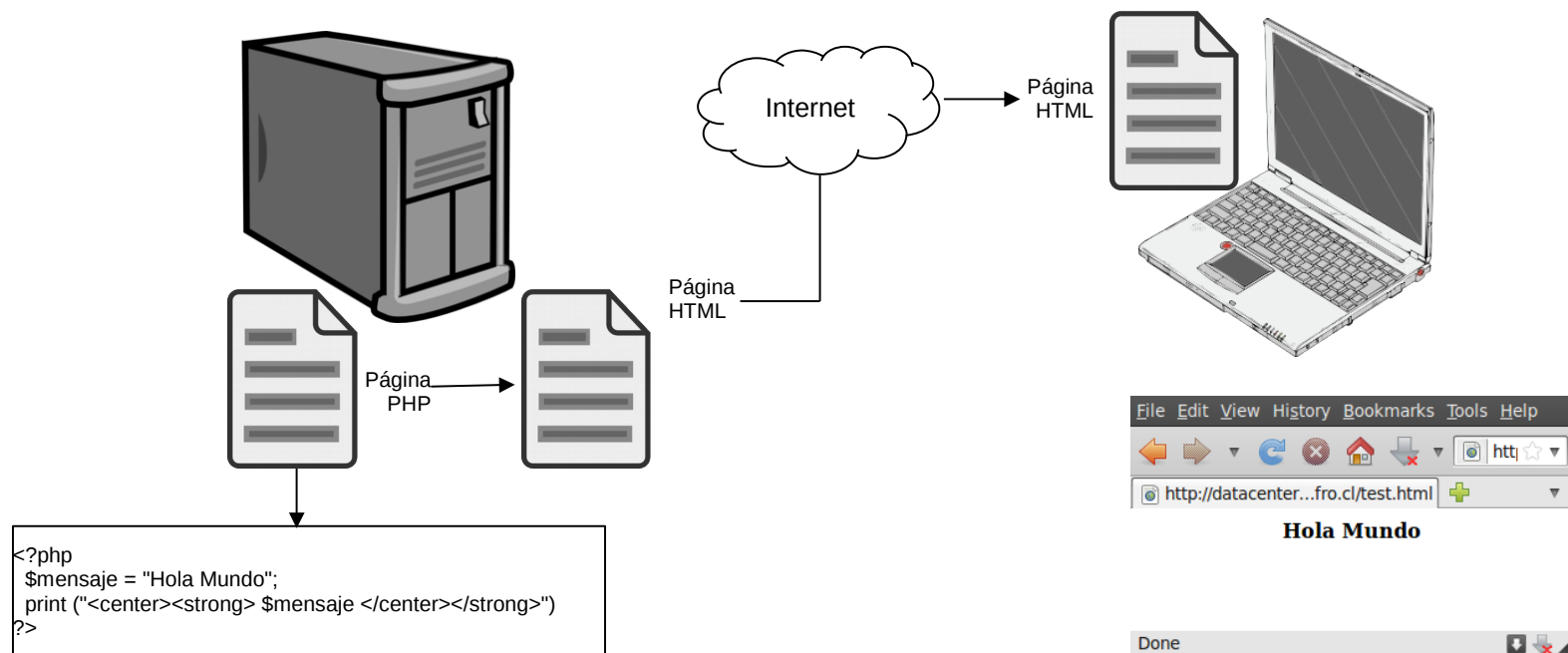
## Introducción

PHP - **Hypertext Preprocessor** es un **lenguaje de script** del lado del servidor. Otros lenguajes similares son Perl, ASP, JSP o ColdFusion. Los scripts PHP son interpretados y ejecutados por el servidor web antes de servir las páginas al cliente. El cliente no ve el código PHP sino los resultados que éste produce.



La mejor forma de visualizar cómo funciona **PHP** es comparándolo con el funcionamiento de **HTML**.

**HTML**, es un lenguaje de **presentación de contenidos web estáticos**. Lo que hace HTML es darle instrucciones al navegador de internet de “cómo debe mostrar el contenido” en el PC del cliente. PHP, por el contrario, toma código dentro de las páginas, lo ejecuta en el servidor mediante un intérprete (generando una página html) y envía el resultado al cliente. El cliente no puede visualizar el código del programa, sólo su resultado.



# Instalación de PHP

---

La instalación típica en entornos GNU/Linux para dar soporte a sitios web dinámicos con PHP implica instalar: el servidor web de Apache, MySQL server y PHP (lo que se conoce como **LAMP** – Linux, Apache, MySQL, PHP).

Ya que se ha instalado Apache y Mysql anteriormente, sólo revisaremos la instalación de PHP. Por repositorios (en CentOS 6.x o cualquier distribución basada en YUM), la instalación es bastante simple:

```
yum install php php-mysql php-common
```

Con esto tendremos las librerías básicas para el soporte PHP y además la conexión entre PHP y MySQL.

# Sintáxis y elementos básicos de PHP

---

1. PHP es sensible a mayúsculas y minúsculas.
2. Las páginas PHP corresponden a archivos con extensión “.php” que se guardan en algún directorio de un servidor web (Apache).
3. Una página PHP siempre debe empezar con “<?php” y terminar con “?>”, para indicar que lo que está escrito entre éstos símbolos son instrucciones PHP):

```
<?php  
... ;  
...Instrucciones en lenguaje PHP;  
... ;  
?>
```

4. Las instrucciones PHP se separan con un “;” (igual que en C++).
5. Para incluir comentarios en PHP se utilizan:

```
// para un comentario de una línea  
/* Para un comentario de  
  Varias Líneas */
```

6. PHP soporta 8 tipos de datos primitivos:  
Tipos escalares: Boolean, Integer, Double, String  
Tipos compuestos: Array, Object  
Tipos especiales: Resource, NULL

El tipo de una variable no se especifica. Se decide en tiempo de ejecución, cuando la variable recibe o se le asigna un valor determinado.

## 7. Variables

Los nombres de variables siempre van precedidos de un “\$”.

El nombre es sensible a mayúsculas y minúsculas.

Por ejemplo:

```
<?php  
$valor=5;  
print (“El Valor es: $valor”);  
?>
```

## Imprimir mensajes

Para generar salidas se utilizan dos comandos: print y echo.

Ámbos comandos funcionan de manera similar, la única diferencia entre un comando y otro es la forma de cómo concatenan los strings: echo usa una “,” en cambio print usa un “.”

Comando	Salida
echo “Hola ”, “mundo”;	Hola mundo
print “Hola “.”mundo”; print (“Hola “.”mundo”); // funciona igual al anterior.	Hola mundo

## Expresiones y Operadores

<b>Operadores aritméticos</b>	Suma: + Resta: - Multiplicación: * División: / Porcentaje: % Incremento en 1: ++ Decremento en 1: --
<b>Operador de asignación</b>	=
<b>Operadores de comparación</b>	Igual: == Distinto: != Menor que: < Mayor que: > Menor o Igual: <= Mayor o Igual >=
<b>Operadores lógicos</b>	And: && Or:

## Estructuras de control

### Estructura selectiva if-else

```
if (condición1)  
{  
    sentencia 1;  
    sentencia 2;  
    ...  
}  
else  
{  
    sentencia 3;  
    sentencia 4;  
}
```

### Estructura repetitiva While

```
While (condición1)  
{  
    sentencia 1;  
    sentencia 2;  
    ...  
}
```

### Estructura repetitiva For

```
For (inicialización; condición; incremento)  
{  
    sentencia 1;  
    sentencia 2;  
    ...  
}
```

Más información sobre comandos y funciones PHP

**w3schools**

<http://www.w3schools.com/php/default.asp>