

Errores vs buenas prácticas en PHP

1.

Errores al momento de la ejecución del código en PHP

Error:

```
<?php
Contenido PHP
?>
```

toda cadena de carácter que contenga espacios deberá

ser escrita entre comillas simples o dobles y toda sentencia debe terminar con un punto y coma (;).

Error de sintaxis por una línea de código que no es una sentencia correcta de PHP

Buena práctica:

```
<?php
echo "Contenido PHP";
?>
```

Muestra: **Contenido PHP**

Resultado exitoso al ejecutar una sentencia válida de PHP

2.

En este caso, el motivo del error es que `print` no es una palabra reservada del lenguaje ni una función (debería decir **`print`**) y cuando encuentra la cadena no sabe cómo seguir.

Error:

```
<?php
print "<p>Hola</p>\n";
?>
```

Parse error: syntax error, unexpected T_CONSTANT_ENCAPSED_STRING in ejemplo.php on line 2

Buena práctica para evitar este error es tener en cuenta como se debe escribir, ya que print en php no es realmente una función (es un constructor de lenguaje), por lo tanto, no es necesario usar paréntesis para indicar su lista de argumentos.

Buena práctica:

```
<?php
print "<p>Hola</p>\n";
?>
```

Muestra: **Hola**

3.

En este caso, el motivo de error es que la primera sentencia no finaliza en punto y coma (;) y no se esperaba el segundo print. Por cierto, la última sentencia de un bloque php puede no finalizar en punto y coma, aunque se recomienda hacerlo. Los paréntesis, llaves y corchetes deben estar emparejados (por cada signo abierto debe hacer uno cerrado).

```
<?php
print "<p>Hola</p>\n"
print "<p>A dios</p>\n"
?>
```

Parse error: syntax error, unexpected T_PRINT in **ejemplo.php** on line **3**

La buena práctica para evitar este error es que debemos tener en cuenta cada una de estas instrucciones debe terminar con un punto y coma (;).

Ejemplo:

```
<?php
print "<p>Hola</p>\n";
print "<p>A dios</p>\n";
?>
```

Muestra: **Hola**

A dios