

Como habrás podido comprobar, en PHP el convenio de uso de nombres de funciones no es nada homogéneo, y ha ido cambiando en el tiempo:

Tenemos por ejemplo:

- `sub_str` (todo minúsculas, separados por `_`)
- `strpos` (todo minúsculas, junto)
- `HTML_Upload_Error` (en PEAR, Mayúsculas y minúsculas, separados por `_`)
- `DateTimeInterface`

La recomendación actual, es utilizar **camelCase** (la primera letra en minúsculas, luego cada palabra en mayúsculas sin separador) para funciones, variables, atributos y métodos; y **PascalCase** (igual pero la primera letra en mayúsculas) para clases.

Si quieres averiguar más información sobre convenciones de código, te interesa revisar la web del FIG (Framework Interoperativity Group), concretamente su estándar PSR-4

<http://www.php-fig.org/http://www.php-fig.org/psr/psr-4/>

En resumen, los espacios de nombres según esta especificación deben ser de esta forma:

```
\<Vendor>\<0-n subnamespaces>\<Class name>
```

Con algunos puntos adicionales:

- Los caracteres `_` no pueden tener ningún significado especial en las clases
- Algunos de los primeros subespacios de nombres (prefijo) tienen que ser asignados a un directorio base
- El resto de los espacios de nombres corresponderán a carpetas dentro del directorio base
- El nombre de la clase debe coincidir con el de un fichero terminado en `“.php”` (sensible a mayúsculas/minúsculas).

La ventaja de seguir estas recomendaciones son dos:

- Podemos utilizar uno o varios frameworks en conjunción con nuestro código sabiendo que no habrá ningún conflicto de nombres
- Podemos utilizar el cargador de clases de otro framework de forma que sea compatible con nuestras clases.

