

1 Curso POO PHP Xestión de erros

1.1 Xestión de erros

PHP define unha **clasificación dos erros** que se poden producir na execución dun programa e ofrece métodos para axustar o tratamento destes. Para facer referencia a cada un dos niveis de erro, PHP define unha serie de constantes. Cada nivel identifícase por unha constante. Por exemplo, a constante **E_NOTICE** fai referencia a avisos que poden indicar un erro ao executar o guión, e a constante **E_ERROR** engloba erros fatais que provocan que se interrompa forzosamente a execución.

A configuración inicial de como se vai tratar cada erro segundo o seu nivel realízase en **php.ini** o ficheiro de configuración de PHP. Entre os principais parámetros que podes axustar están:

- **error_reporting**. Indica que tipos de erros se notificarán. O seu valor fórmase utilizando os operadores a nivel de bit para combinar as constantes anteriores. Por exemplo, o valor "**E_ALL & ~E_NOTICE**" indica que se notifiquen todos os erros (**E_ALL**) salvo os avisos en tempo de execución (**E_NOTICE**). Cando non se pasan parámetros, devolve o nivel de notificación actual.
- **display_errors**. No seu valor por defecto (1), fai que as mensaxes se envíen á saída estándar (e polo tanto móstrense no navegador). Débese desactivar (0) nos servidores que non se usan para desenvolvemento senón para produción.

Dende código, podes usar a función **error_reporting** coas constantes anteriores para establecer o nivel de notificación nun momento determinado. Por exemplo, se nalgún lugar do teu código figura unha división na que exista a posibilidade de que o divisor sexa cero, cando isto aconteza obterás unha mensaxe de erro. Para evitalo, podes desactivar a notificación de erros de nivel **E_WARNING** antes da división e restaurala ao seu valor normal a continuación:

```
nivel_erros = error_reporting();
error_reporting = (nivel_erros & ~E_WARNING);
$resultado = $dividendo / $divisor;
// Recuperamos o nivel anterior
error_reporting(nivel_erros);
```

Aínda que en moitos casos é máis sinxelo empregar o operador de supresión de erros @:

```
@$resultado = $dividendo / $divisor;
```

Para poder rexistrar a un arquivo de texto os erros que se produzan en un sitio web en produción, debemos asegurarnos de ter activa en **php.ini** a directiva **log_errors** e indicar empregando **error_log** o ficheiro a empregar como rexistro.

```
log_errors = On
error_log = "\xampp\php\logs\php_error_log"
```

1.1.1 Manexador de erros a medida

Para obter máis control sobre os erros que se poidan producir, existe tamén a posibilidade de substituír a xestión destes pola que ti definas. É dicir, podes programar unha función para que sexa a que executa PHP cada vez que se produce un erro. O nome desa función indícase utilizando **set_error_handler** e debe ter como mínimo dous parámetros obrigatorios (o nivel do erro e a mensaxe descritiva) e ata outros tres opcionais con información adicional sobre o erro (o nome do ficheiro en que se produce, o número de liña, e o estado das variables nese momento).

```
set_error_handler("meuXestorDeErros");
$resultado = $dividendo / $divisor;
restore_error_handler();

function meuXestorDeErros($nivel, $mensaxe)
{
    switch($nivel) {
        case E_WARNING:
            echo "Erro de tipo WARNING: $mensaxe.<br />";
            break;
        default:
            echo "Erro de tipo non especificado: $mensaxe.<br />";
    }
}
```

A función **restore_error_handler** restaura o manexador de erros orixinal de PHP (máis concretamente, o que se estaba a usar antes da chamada a **set_error_handler**).

--V ctor Lourido 19:58 13 jul 2013 (CEST)