

1 Curso POO PHP Clases e métodos abstractos

1.1 Clases e métodos abstractos

A partires de PHP5 introducíronse na linguaxe as clases e os métodos abstractos.

As **clases abstractas** son semellantes aos interfaces en que a partires delas non se poden instanciar obxectos. Estas clases poden ter dous tipos de métodos:

- **Métodos regulares**, que inclúen o seu código, semellantes aos que xa vimos.
- **Métodos abstractos**, que do mesmo xeito que nos interfaces, declaran soamente a estrutura do método pero non inclúen código. Estes métodos deben ser implementados obrigatoriamente por calquera clase non abstracta que herde da clase abstracta.

Calquera clase que defina un método abstracto debe ser declarada como abstracta (é lóxico pois non ten implementado algún método e por tanto non se poderán instanciar obxectos a partires dela).

Por exemplo, se non imos traballar con obxectos da clase Produto, poderíamos facela abstracta e empregala como base para definir os métodos comúns ás clases que deriven dela:

```
abstract class Produto {  
    public $codigo;  
    public $nome;  
    public $nome_corto;  
    public $PVP;  
  
    public function mostra_codigo() {  
        print "<p>" . $this->codigo . "</p>";  
    }  
  
    // O seguinte método deberá ser implementado nas clases que deriven de Produto  
    abstract public function mostra_producto();  
}
```

Outra diferenza entre as clases abstractas e os interfaces é que, tal e como acabas de ver, as primeiras poden conter atributos e os interfaces non.

En canto á herdanza múltiple, as clases abstractas compórtanse como calquera outra clase. Isto é, non se permite crear unha clase que herde de máis de unha clase, sexa ou non abstracta. Esta é unha das principais vantaxes dos interfaces sobre as clases abstractas, pois como xa vimos é posible crear unha clase que implemente dous ou máis interfaces.

--Víctor Lourido 22:49 12 jul 2013 (CEST)