

# クラス



クラス



# PHPでのクラス定義

- PHPでもクラスを利用したオブジェクト指向開発が可能です
- 一般的にPHPの開発は、共通的な処理はクラス化して再利用していく手法が多くみられます



# PHPのクラス定義

```
Class クラス名 { ← classの後にクラス名を記載
    private $変数名 = ""; ← public、private等の後に変数名を定義

    public function メソッド名($引数名,$引数名) {
        :
    } ← public、private等の後に通常通りの関数の書き方
}
```



# PHPのクラスのnew

```
$cls = new ClsA();
```

ClsA型のクラスを作るという意味（Javaのnewと同じ）



# PHPのクラスのメソッドを利用

```
$cls = new ClsA();  
$cls->func1();
```

ClsA型のメソッドfunc1を呼び出す



# クラスを別ファイルに定義

- 共通的な処理をクラス化した場合、そのクラスを複数ファイル(ページ)で利用することが考えられます
- その場合は、クラスを別ファイルとして定義し、利用するファイル内に読み込むという方法を使います



# クラスのファイル化

下記のような内容を記載したファイルを作成する

```
<?php
Class ClsA{
    private $st = "";

    public function func1(){
        :
    }
}
?>
```

clsA.phpというファイル名で保存





# クラスのファイルを読み込む

クラスを利用するファイルを読み込む

```
<?php
require 'clsa.php'; ←clsa.php(クラスを定義したファイル名)を指定して読み込み
:
?>
```

※require 'clsa.php' ; (ファイルを読み込み、読み込めなければエラー発生)

require\_once 'clsa.php' ; (ファイルを1度だけ読み込み、読み込めないときは処理中止)



# Webフォルダ内に「enshu9」フォルダを作成

今回作成するプログラムは、このフォルダ内に作成します。



# クラスを作成する

- 下記の関数をもつクラス(Cls9.php)を作成し、その関数を利用してみましょう
- 関数:任意の言葉を渡すと、その中に「い」があった場合、その部分を  
「いえーい！」に変換して文字列を返す

**※文字を変換する方法はネットで検索**すること

- 入力画面にテキストボックスと送信ボタンを用意する(enshu9\_input.php)
- 出力画面で、入力画面から渡された文字を、上記で作成したクラスに渡し、  
結果を画面に表示する(enshu9\_output.php)

