# Dojo Verlan

**travail sur** : manipulation de chaine, conversion, tableaux, conditions

Le but du dojo est de transformer certains mots d’une phrase en permutant les deux moitiés du mot, à condition que le mot fasse au moins 6 caractères.

En entrée, la phrase à transformer.

En sortie, la phrase transformée.  
  
Ex : La maison près de la fontaine => la sonmai près de la ainefont  
  
Optionnel : la césure devra se faire sur la consonne au plus proche du centre du mot  
  
Ex : la maison près de la fontaine => la sonmai près de la tainefon

## Fonction

class KataVerlan  
{  
 const CONSONNANT = "bcdfghjklmpqrstvwxz";  
 public static function action($value) // simple version  
 {  
 $strWords = explode(" ", $value);  
 $newSentence = "";  
 foreach($strWords as $word){  
 $size = strlen($word);  
 if ($size >5){  
 $center = (int)($size / 2);  
 $left = substr($word, 0, $center);  
 $right = substr($word, $center, ($size - $center));  
 $newSentence .= $right . $left . " ";  
 } else {  
 $newSentence .= $word . " ";  
 }  
 }  
 return trim($newSentence);  
 }  
 public static function actionOption($value) // more complex version  
 {  
 $strWords = explode(" ", $value);  
 $newSentence = "";  
 foreach($strWords as $word){  
 $size = strlen($word);  
 if ($size >5){  
 $center = self::getBestCenter($word, $size);  
 $left = substr($word, 0, $center);  
 $right = substr($word, $center, ($size - $center));  
 $newSentence .= $right . $left . " ";  
 } else {  
 $newSentence .= $word . " ";  
 }  
 }  
 return trim($newSentence);  
 }  
 private static function getBestCenter($word, $size){  
 $center = (int)($size / 2);  
 $increment = 0;  
 $phase = 1;  
 while (false === strpos(self::CONSONNANT, substr($word, ($center - 1) + $increment, 1))){  
 switch($phase){  
 case 1:  
 $increment++;  
 $phase++;  
 break;  
 case 2:  
 $increment = -$increment;  
 $phase++;  
 break;  
 case 3:  
 $increment = abs($increment);  
 $phase = 1;  
 break;  
 }  
 }  
 return $center + $increment - 1;  
 }  
}

## Tests

public function testAction()  
 {  
 $this->assertEquals(‘route’, \wcs\KataExemple::action("route"));

$this->assertEquals(‘maison’, \wcs\KataExemple::action("sonmai"));

$this->assertEquals(‘La maison près de la fontaine’,

\wcs\KataExemple::action("La sonmai près de la ainefont"));

}

public function testOptionAction()  
 {  
 $this->assertEquals(‘route’, \wcs\KataExemple::action("route"));

$this->assertEquals(‘maison’, \wcs\KataExemple::action("sonmai"));

$this->assertEquals(‘La maison près de la fontaine’,

\wcs\KataExemple::action("La sonmai près de la tainefon"));

}