

Méthodes de classe (statiques)

```

1<?php
2class Alouette {

3/** Méthode titre
4* Retourne le titre tel qu'affiché dans le head
  et le h1 de la page
5* @param string $oiseau - Le nom de l'oiseau
  (Alouette)
6* @param string $qualite - La qualité que l'on
  donne à l'oiseau (gentille)
7* @return string
8*/
- 9static public function titre($oiseau, $qualite) {
- 10return $oiseau.', '.$qualite.' '.$oiseau;
11}
12/** Méthode chanson
13* Retourne la chanson au complet dans un
  div.chanson
18* @uses strophe
19* @return string
20*/
- 21static public function chanson($oiseau, $qualite,
  $action, $membres) {
22$resultat = '';
- 23$membresDits = array();
24$resultat .= '<div class="chanson">';
- 25foreach ($membres as $membre) {
26array_unshift($membresDits, $membre);
- 27$resultat .= Alouette::strophe($oiseau, $qualite,
  $action, $membre, $membresDits);
28}
29$resultat .= '</div>';
30return $resultat;
40* @uses couplet
41* @return string
42*/
- 43static public function strophe($oiseau, $qualite,
  $action, $membre, $membresDits) {
44$resultat = '';
45$resultat .= '<div class="strophe">';
- 46$resultat .= self::refrain($oiseau, $qualite,
  $action);
- 47$resultat .= self::couplet($oiseau, $action,
  $membre, $membresDits);
48$resultat .= '</div>';
49return $resultat;
50}
58* @uses appelReponse
59* @return string
60*/
- 61static public function refrain($oiseau, $qualite,
  $action) {
62$resultat = '';
- 63$resultat .= '<div>'.self::titre($oiseau,
  $qualite).'</div>';
- 64$resultat .= '<div>'.$oiseau.', je
  '.$action.'.</div>';
- 65$resultat = self::appelReponse($resultat);
66$resultat = '<div
  class="refrain">'.$resultat.'</div>';
67return $resultat;
68}

```

Méthodes d'objet (non statiques)

```

1<?php
2class Alouette {
+ 3private $oiseau = "";
+ 4private $qualite = "";
+ 5private $action = "";
+ 6private $membres = [];
+ 7public function __construct($oiseau, $qualite,
  $action, $membres) {
+ 8$this->oiseau = $oiseau;
+ 9$this->qualite = $qualite;
+ 10$this->action = $action;
+ 11$this->membres = $membres;
+ 12}
13/** Méthode titre
14* Retourne le titre tel qu'affiché dans le head
  et le h1 de la page
15* @param string $oiseau - Le nom de l'oiseau
  (Alouette)
16* @param string $qualite - La qualité que l'on
  donne à l'oiseau (gentille)
17* @return string
18*/
+ 19public function titre() {
+ 20return $this->oiseau.', '.$this->qualite.'
  '.$this->oiseau;
21}
22/** Méthode chanson
23* Retourne la chanson au complet dans un
  div.chanson
28* @uses strophe
29* @return string
30*/
+ 31public function chanson() {
32$resultat = '';
+ 33$membresDits = [];
34$resultat .= '<div class="chanson">';
+ 35foreach ($this->membres as $membre) {
36array_unshift($membresDits, $membre);
+ 37$resultat .= $this->strophe($membre,
  $membresDits);
38}
39$resultat .= '</div>';
40return $resultat;
50* @uses couplet
51* @return string
52*/
+ 53public function strophe($membre, $membresDits) {
54$resultat = '';
55$resultat .= '<div class="strophe">';
+ 56$resultat .= $this->refrain();
+ 57$resultat .= $this->couplet($membre,
  $membresDits);
58$resultat .= '</div>';
59return $resultat;
60}
68* @uses appelReponse
69* @return string
70*/
+ 71public function refrain() {
72$resultat = '';
+ 73$resultat .= '<div>'. $this->titre().'</div>';
+ 74$resultat .= '<div>'. $this->oiseau.', je
  '.$this->action.'.</div>';
+ 75$resultat = $this->appelReponse($resultat);
76$resultat = '<div
  class="refrain">'.$resultat.'</div>';
77return $resultat;
78}

```

Méthodes de classe (statiques)

```

78 * @return string
79 * @note L'appel est dans un div.appel et la
   réponse, dans un div.reponse
80 */
- 81 static public function couplet($oiseau, $action,
   $membre, $membresDits) {
82 $resultat = '';
- 83 $resultat .= self::actionMembre($action,
   $membre);
- 84 $resultat .= self::enumMembres($membresDits);
- 85 $resultat .= self::appelReponse($oiseau);
86 $resultat .= '<div>Aaaah . . . </div>';
87 return $resultat;
88 }
92 * @return string
93 * @note L'appel est dans un div.appel et la
   réponse, dans un div.reponse
94 */
- 95 static public function appelReponse($phrase) {
96 $resultat = '';
97 $resultat .= '<div
   class="appel">'. $phrase. '</div>';
98 $resultat .= '<div
   class="reponse">'. $phrase. '</div>';
104 * @uses appelReponse
105 * @return string
106 */
- 107 static public function enumMembres($membres) {
108 $resultat = '';
109 foreach ($membres as $membre) {
- 110 $resultat .= self::appelReponse('Et ' . $membre);
111 }
112 return $resultat;
113 }
118 * @uses appelReponse
119 * @return string
120 */
- 121 static public function actionMembre($action,
   $membre) {
- 122 $resultat = self::appelReponse('Je ' . $action. '
   ' . $membre);
123 return $resultat;
124 }
- 125 static public function form($donnees) {
126 $resultat = '';
127 $resultat .= '<form action="perso.php"
   action="get">';
128 $resultat .= '<div class="champ">';
129 $resultat .= '<label
   for="oiseau">Oiseau</label>';
- 130 $resultat .= '<input type="text" name="oiseau"
   id="oiseau" value="'. $donnees['oiseau'] . '"/>';
131 $resultat .= '</div>';
132 $resultat .= '<div class="champ">';
133 $resultat .= '<label
   for="qualite">Qualité</label>';
- 134 $resultat .= '<input type="text" name="qualite"
   id="qualite" value="'. $donnees['qualite'] . '"/>';
135 $resultat .= '</div>';
136 $resultat .= '<div class="champ">';
137 $resultat .= '<label
   for="action">Action</label>';
- 138 $resultat .= '<input type="text" name="action"
   id="action" value="'. $donnees['action'] . '"/>';
139 $resultat .= '</div>';
140 $resultat .= '<div class="champ">';
141 $resultat .= '<label
   for="membres">Membres</label>';
142 // $membres = '';
- 143 // foreach($donnees['membres'] as $id => $membre)
   {
144 // $membres .= $membre. "\r\n";
145 // }

```

Méthodes d'objet (non statiques)

```

88 * @return string
89 * @note L'appel est dans un div.appel et la
   réponse, dans un div.reponse
90 */
+ 91 public function couplet($membre, $membresDits) {
92 $resultat = '';
+ 93 $resultat .= $this->actionMembre($membre);
+ 94 $resultat .= $this->enumMembres($membresDits);
+ 95 $resultat .= $this->appelReponse($this->oiseau);
96 $resultat .= '<div>Aaaah . . . </div>';
97 return $resultat;
98 }
102 * @return string
103 * @note L'appel est dans un div.appel et la
   réponse, dans un div.reponse
104 */
+ 105 public function appelReponse($phrase) {
106 $resultat = '';
107 $resultat .= '<div
   class="appel">'. $phrase. '</div>';
108 $resultat .= '<div
   class="reponse">'. $phrase. '</div>';
114 * @uses appelReponse
115 * @return string
116 */
+ 117 public function enumMembres($membres) {
118 $resultat = '';
119 foreach ($membres as $membre) {
+ 120 $resultat .= $this->appelReponse('Et ' . $membre);
121 }
122 return $resultat;
123 }
128 * @uses appelReponse
129 * @return string
130 */
+ 131 public function actionMembre($membre) {
+ 132 $resultat = $this->appelReponse('Je
   ' . $this->action. ' ' . $membre);
133 return $resultat;
134 }
+ 135 public function form() {
136 $resultat = '';
137 $resultat .= '<form action="perso.php"
   action="get">';
138 $resultat .= '<div class="champ">';
139 $resultat .= '<label
   for="oiseau">Oiseau</label>';
+ 140 $resultat .= '<input type="text" name="oiseau"
   id="oiseau" value="'. $this->oiseau . '"/>';
141 $resultat .= '</div>';
142 $resultat .= '<div class="champ">';
143 $resultat .= '<label
   for="qualite">Qualité</label>';
+ 144 $resultat .= '<input type="text" name="qualite"
   id="qualite" value="'. $this->qualite . '"/>';
145 $resultat .= '</div>';
146 $resultat .= '<div class="champ">';
147 $resultat .= '<label
   for="action">Action</label>';
+ 148 $resultat .= '<input type="text" name="action"
   id="action" value="'. $this->action . '"/>';
149 $resultat .= '</div>';
150 $resultat .= '<div class="champ">';
151 $resultat .= '<label
   for="membres">Membres</label>';
152 // $membres = '';
+ 153 // foreach($this->membres as $id => $membre) {
154 // $membres .= $membre. "\r\n";
155 // }

```

Méthodes de classe (statiques)

```
- 146 $membres = implode("\r\n", $donnees['membres']);  
147 $resultat .= '<textarea name="membres"  
    id="membres" cols="20"  
    rows="10">'.$membres.'</textarea>';  
148 $resultat .= '</div>';  
149 $resultat .= '<div class="submit"><input  
    type="submit" value="Composer"/></div>';
```

Méthodes d'objet (non statiques)

```
+ 156 $membres = implode("\r\n", $this->membres);  
157 $resultat .= '<textarea name="membres"  
    id="membres" cols="20"  
    rows="10">'.$membres.'</textarea>';  
158 $resultat .= '</div>';  
159 $resultat .= '<div class="submit"><input  
    type="submit" value="Composer"/></div>';
```