Table des matières

[1. void afficherTableau($tab) 1](#_Toc5888621)

[Lignes à coller dans le main 1](#_Toc5888622)

[2. tab[] coderMot($mot) 1](#_Toc5888623)

[Lignes à coller dans le main 2](#_Toc5888624)

[3. int[] testerLettre($lettre, $tab,$depart) 2](#_Toc5888625)

[Lignes à coller dans le main 2](#_Toc5888626)

[4. tab[] ajouterUneLettre($lettre, $tab, $position) 2](#_Toc5888627)

[Lignes à coller dans le main 2](#_Toc5888628)

[5. tab[] ajouterLesLettres($lettre, $ tab, $listePosition) 2](#_Toc5888629)

[Lignes à coller dans le main 2](#_Toc5888630)

[6. void afficherMauvaisesLettres($listeLettres) 2](#_Toc5888631)

[Lignes à coller dans le main 3](#_Toc5888632)

[7. void DessinerPendu($nbErreur) 3](#_Toc5888633)

[8. string choisirMot() 4](#_Toc5888634)

[9. char DemanderLettre() 4](#_Toc5888635)

[Lignes à coller dans le main 4](#_Toc5888636)

[10. $res TesterGagner($nberreur, $tab) 4](#_Toc5888637)

[Lignes à coller dans le main 4](#_Toc5888638)

[11. LancerPartie() ; 5](#_Toc5888639)

# void afficherTableau($tab)

// méthode qui permet d'afficher le tableau passé en paramètre et de séparer les lettres par des espaces

## Lignes à coller dans le main

$t = array( 'B', 'O', 'N', 'J', 'O', 'U', 'R' );

echo "Cette méthode doit donner B O N J O U R et ca donne : " ;

afficherTableau($t);

# tab[] coderMot($mot)

// méthode qui prend un mot en paramètre d'entrée et qui renvoi un tableau de caractères contenant autant de case que de lettres dans le mot. Chacune de ces cases contient un \_

## Lignes à coller dans le main

$test = "bonjour";

echo "Cette méthode doit donner \_ \_ \_ \_ \_ \_ \_ et ca donne : " . afficherTableau(coderMot($test));

# int[] testerLettre($lettre, $tab,$depart)

// méthode qui cherche toutes les occurrences d'une lettre passée en paramètre dans un tableau de caractères passé aussi en paramètre. Cette méthode retourne toutes les positions dans un tableau

$depart represente le point de depart de la recherche

## Lignes à coller dans le main

Echo "Cette méthode doit donner \n 1 \n 4 et ca donne \n" ;

$t = array( 'B', 'O', 'N', 'J', 'O', 'U', 'R' );

$positions = testerLettre('O', $t,0);

foreach ($positions as $pos)

{

echo("position : ".$pos."\n");

}

# tab[] ajouterUneLettre($lettre, $tab, $position)

//méthode qui modifie le tableau passé en paramètre en affectant la lettre à la position passée en paramètre

## Lignes à coller dans le main

Echo "Cette méthode doit donner B O N K O U R et ca donne";

$t = array( 'B', 'O', 'N', 'J', 'O', 'U', 'R' );

afficherTableau( ajouterUneLettre('K', $t, 3));

# tab[] ajouterLesLettres($lettre, $ tab, $listePosition)

//méthode qui appelle la méthode ajouterUneLettre pour toutes les valeurs contenues dans la liste en paramètre

## Lignes à coller dans le main

$motATrouver="BONJOUR";

$t = array( 'B', '\_', 'N', 'J', '\_', 'U', '\_' );

echo "Cette méthode doit donner B O N J O U \_ et ca donne ";

afficherTableau(ajouterLesLettres('O', $t, testerLettre('O', str\_split($motATrouver),0)));

// en l’absence des autres methodes

Print\_r(ajouterLesLettres('O', $t,[1,4]);

# void afficherMauvaisesLettres($listeLettres)

// méthode qui permet d'afficher les caractères contenus dans la liste passée en paramètre à l'écran

## Lignes à coller dans le main

$liste = array('A','B','C') ;

echo "Cette méthode doit donner :\n Les lettres non présentes sont A,B,C \n et ca donne \n" ;

afficherMauvaisesLettres($liste);

# void DessinerPendu($nbErreur)

[\\méthode](file:///\\méthode) qui permet de dessiner le pendu en fonction du nombre d’erreur

function DessinerPendu($nbErreur)

{

switch ($nbErreur)

{

case 0:

Echo " "."\n";

Echo " "."\n";

Echo " "."\n";

Echo " "."\n";

Echo " "."\n";

Echo " "."\n";

Echo " "."\n";

Echo " "."\n";

break;

case 1:

Echo " "."\n";

Echo " "."\n";

Echo " "."\n";

Echo " "."\n";

Echo " "."\n";

Echo " "."\n";

Echo " "."\n";

Echo " \_\_\_\_\_\_\_\_ "."\n";

break;

case 2:

Echo " "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " \_|\_\_\_\_\_\_\_ "."\n";

break;

case 3:

Echo " \_\_\_\_\_\_\_\_ "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " \_|\_\_\_\_\_\_\_ "."\n";

break;

case 4:

Echo " \_\_\_\_\_\_\_\_ "."\n";

Echo " | | "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " \_|\_\_\_\_\_\_\_ "."\n";

break;

case 5:

Echo " \_\_\_\_\_\_\_\_ "."\n";

Echo " | | "."\n";

Echo " | O "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " \_|\_\_\_\_\_\_\_ "."\n";

break;

case 6:

Echo " \_\_\_\_\_\_\_\_ "."\n";

Echo " | | "."\n";

Echo " | O "."\n";

Echo " | | "."\n";

Echo " | | "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " \_|\_\_\_\_\_\_\_ "."\n";

break;

case 7:

Echo " \_\_\_\_\_\_\_\_ "."\n";

Echo " | | "."\n";

Echo " | O "."\n";

Echo " | /|\\ "."\n";

Echo " | | "."\n";

Echo " | "."\n";

Echo " | "."\n";

Echo " \_|\_\_\_\_\_\_\_ "."\n";

break;

case 8:

Echo " \_\_\_\_\_\_\_\_ "."\n";

Echo " | | "."\n";

Echo " | O "."\n";

Echo " | /|\\ "."\n";

Echo " | | "."\n";

Echo " | / \\ "."\n";

Echo " | "."\n";

Echo " \_|\_\_\_\_\_\_\_ "."\n";

break;

default:

break;

}

}

# string choisirMot()

[\\ méthode](file:///\\méthode) qui renvoi un mot en le choisissant au hasard parmi une liste de mots

Utiliser la méthode CreerDico disponible dans le dossier

# char DemanderLettre()

// méthode qui demande une lettre à l’utilisateur, elle vérifie que le caractère saisi est une lettre et le renvoi en majuscule.

## Lignes à coller dans le main

$c = DemanderLettre();

echo $c;

# $res TesterGagner($nberreur, $tab)

//méthode qui renvoi 1 si la partie est gagné, -1 si la partie est perdu, 0 si la partie continue. Elle reçoit en paramètre le nombre d’erreurs et le tableau contenant le mot composé

## Lignes à coller dans le main

$t = array( 'B', '\_', 'N', 'J', 'O', 'U', 'R' );

Echo "Cette méthode doit donner -1 et ca donne " . testerGagner(9, $t)."\n";

Echo "Cette méthode doit donner 0 et ca donne " . testerGagner(3, $t)."\n";

$t[1] = 'O' ;

Echo "Cette méthode doit donner 1 et ca donne " . testerGagner(2, $t)."\n";

# LancerPartie() ;

//méthode qui lance et gère une partie