

```

<?php
//-----//
//-----Partie 1. Algorithme-----//
//-----//

//Le langage de programmation utilisé dans cette partie est le PHP.
//-----Question 1-----//

/*
Question 1
Donner les valeurs de $a, $b, $c.
Réponses:
$a = 2
$b = 10
$c = 7

$a = 10
$b = 10
$c = 1
*/

/*
$a = 2;
$b = 4 + 3 * $a;
$c = $a + $b / 2;

$a = 10;
$c = $b - ($a - 1);

echo $c;
*/

```

//-----Question 2-----//

/*

Question 2

Donner la valeur de \$b à la fin de cette méthode pour \$c valant respectivement 1, 5 et 45. Réponses:

Réponses:

Quand \$c = 1 alors \$b = 4

Quand \$c = 5 alors \$b = 5

Quand \$c = 45 alors \$b = 45

```
function myFunction($c)
{
    $b = $c;
    for ($i = 0; $i < ($c % 5) * $c; $i++) {
        if ($i % 3 == 1) {
            $b -= 1;
        }
        if ($i % 3 == 2) {
            $b *= 2;
        }
        if ($i % 3 == 0) {
            $b += 3;
        }
    }
    return $b;
}
*/
```

```
//myFunction($c = 45);
```

//-----Question 3-----//

/*

Question 3

* Donner les valeurs de \$a et \$b à la fin de cet algorithme.

*Réponses:

"Rouge" -> \$a = -3 et \$b = 8

"Bleu" -> \$a = -3 et \$b = 9

"Jaune" -> \$a = -3 et \$b = 10

"Vert" -> \$a = -3 et \$b = 11

"Violet" -> \$a = -3 et \$b = 12

"Blanc" -> \$a = -3 et \$b = 13

"Noir" -> \$a = -6 et \$b = 14

"Gris" -> \$a = -6 et \$b = 15

"Marron" -> \$a = -6 et \$b = 16

"Rose" -> \$a = -23 et \$b = 17

*/

\$string = 'Orange';

\$char = 'Y';

function startWith(\$string, \$char)

{ // Function return true if \$char is present in \$string

 \$pos = strpos(\$string, \$char);

 if (\$pos === false) {

 /*

 echo " '\$char' ne se trouve pas dans '\$string'.";

 echo "
";

 */

 return false;

 } else {

 /*

 echo " '\$char' a été trouvée dans '\$string'.";

 echo "
";

 */

 return true;

 }

}

function existChar(\$string, \$char)

{ // Function return true if \$string start with \$char

 \$pos = strpos(\$string, \$char);

 if (\$pos === true) {

```

    /*
    echo "'$char' débute à la position $pos.";
    echo "<br>";
    */
    return true;
} else {
    /*
    echo "'$char' ne se trouve pas dans '$string'.";
    echo "<br>";
    */
    return false;
}
}

```

```

$a = 5;
$b = 7;

```

```

$colors = array(
    "Rouge", "Bleu", "Jaune", "Vert", "Violet",
    "Blanc", "Noir", "Gris", "Marron", "Rose",
);
for ($i = 0; $i < count($colors); $i++) {
    $color = $colors[$i];
    if (startsWith($color, "B")) {
        $a *= $i;
    }
    $i++;
}
foreach ($colors as $color) {
    $b++;
    if (startsWith($color, "R")) {
        $a -= $b;
    }
    if (startsWith($color, "N")) {
        $a = $a * 2;
    }
    if (existChar($color, "n")) {
        $a += 2;
    }
    echo "<br><br>", $a, "<br><br>", $b, "<br><br>";
}

```

```
<?php
```

```
//-----Question 4-----//
```

```
/*
```

Question 4

Donner les valeurs de \$A0, \$A1, \$A2 et \$A3 à la fin de cet algorithme.

Réponses:

\$A0 = A(-1); retourne 1

\$A1 = A(true); retourne 1

\$A2 = A(false); ne retourne rien du tout (null)

\$A3 = A(0); ne retourne rien du tout (null)

```
function A($b)
```

```
{
```

```
    static $a = false;
```

```
    return $b || $a;
```

```
}
```

```
$A0 = A(-1);
```

```
$A1 = A(true);
```

```
$A2 = A(false);
```

```
$A3 = A(0);
```

```
//echo "<br><br>", $A0, "<br><br>", $A1, "<br><br>";
```

```
//echo "<br><br>", $A2, "<br><br>", $A3, "<br><br>";
```

```
echo $A3;
```

```
*/
```

//-----Question 5-----//

/*

Question 5

Donner les valeurs retour de cette méthode avec comme paramètre \$a valant respectivement 5, 11

Réponses:

$a = 5 \% 10 = 5$

$a = 11 \% 10 = 1$

$a = 5;$

```
function Factorial($a = 5)
{
    $a = $a % 10;
    if ($a == 0) {
        return 1;
    }
    return $a * Factorial($a - 1);
}
```

//-----Question 6-----//

/*

Question 6

En reprenant la fonction précédente, est-il possible que la valeur retour soit égale à 0 ?

Réponses:

Si $a = 0$ alors la valeur de retour sera égale à 0.

*/

//-----Question 7-----//

/*

Question 7

En reprenant la fonction précédente, est-il possible de partir en boucle infini ?

Réponses:

En reprenant la fonction précédente, il n'est pas possible de partir en boucle infini car la factorielle de a est le produit des nombres entiers strictement positifs inférieurs ou égaux à a

*/

//-----Question 8-----//

/*

Question 8

Donner la valeur des variables \$A1, \$A2, \$A3 et \$A4

Réponses:

\$A1 = 0

\$A2 = 1

\$A3 = 2

\$A4 = 3

```
function test(&$a)
```

```
{
```

```
    return $a++;
```

```
}
```

```
$A1 = 0;
```

```
$A2 = test($A1);
```

```
$A3 = test($A2);
```

```
$A4 = test($A3);
```

```
*/
```

//-----Question 9-----//

/*

Question 9

Donner la valeur des variables \$A1, \$A2, \$A3 et \$A4

Réponses:

\$A1 = 1

\$A2 = 2

\$A3 = 3

\$A4 = 4

```
function test(&$a)
```

```
{
```

```
    return ++$a;
```

```
}
```

```
$A1 = 0;
```

```
$A2 = test($A1);
```

```
$A3 = test($A2);
```

```
$A4 = test($A3);
```

```
*/
```

```
//-----//
//-----Partie 3. La base de donnée-----//
//-----//
```

```
//-----Question 12-----//
```

```
/*
```

Question 12

Donner les résultats des requêtes suivantes:

Réponses:

SELECT COUNT(*) FROM Customer

= nombre de ligne de la table customer

SELECT COUNT(*) FROM Customer WHERE id LIKE '%0'

= nombre de ligne ou l'id se termine par un 0

SELECT COUNT(*) FROM Customer WHERE domain = 'gmail' OR firstname LIKE 'T%'

=nombre de ligne ou le domaine c'est gmail ou le firstname commence par un T

SELECT COUNT(*) FROM Customer WHERE (id > 0 OR id < 10) AND domain = 'gmail'

=nombre de ligne ou " id superieur a 0 et inferieur a 10 et en meme temp le domain c gmail"

SELECT domain FROM Customer ORDER BY 1 LIMIT 5

ordonné toute la table selon la 1ere colonne

et apres affiché que les 5premiere lignes

SELECT COUNT(*) FROM Access

=nombre de ligne de la table acces

SELECT COUNT(*) FROM Access WHERE status IN('accepted', 'refused')

=nombre de ligne ou le statut est accepted ou refused

SELECT COUNT(*) FROM Access WHERE country = 'Turkey' OR customer_id = 0

=nombre ligne ou country c'est la turkey ou bien le customer_id = 0

SELECT COUNT(*) FROM Access WHERE date > '2016-04-01' AND date < '2016-04-02'

=nombre ligne ou la date est sup a 01/04 et inferieus 02/04

```
*/
```

```
//-----Question 13-----//
```

```
/*
```

Domain	Count	Count2
Free	1	1
Gmail	2	2
Hotmail	1	1
Gmail	6	6
Hotmail	2	2
Localhost	2	2
Yahoo	9	9

```
*/
```


//-----Question 14-----//

/*

Question 14

Donner le résultat de la requête suivante:

```
SELECT
customer.domaine as domain,
COUNT(customer.id) as count,
COUNT(access.id) as count2
FROM
Customer customer
LEFT OUTER JOIN Access access ON (
access.customer_id = customer.id
)
GROUP BY
1
ORDER BY
1, 2, 3
```

Domain	Count	Count 2
boulangier	1	0
free	1	1
gmail	2	2
hotmail	1	1
gmail	7	6
hotmail	2	2
localhost	2	2
yahoo	9	9

/*

//-----Question 15-----//

/*Question 15: Sur une table de la base de données SQL contenant plusieurs centaines de milliers d'enregistrements, nous constatons un temps de réponse de plusieurs secondes, lorsque nous effectuons une requêtes spécifique. Quels moyens / procédure / outils, mettriez-vous en place pour déboguer puis optimiser ces ralentissements ?

Séparer les tables,

Eviter les doublons,

Utiliser les jointure,

Faire des tables intermédiaires,

Simplifier les requêtes