

- [Aucun ami connecté](#)
- [939 Zéros connectés](#)
- [436 273 Zéros inscrits](#)
- 
- [Gya](#)
 - [Avatar](#)
 - [Profil](#)
 - [Identifiants](#)
 - [Études](#)
 - [Réglages](#)
 - [Contacts](#)
 - [Commandes](#)
 -
- [Lisez vos Messages Personnels](#)

Correction du Q.C.M.

[<= Retour au cours](#)

-

Question

Quelle est la classe de base pour créer une exception ?

- Bonne réponse

Exception

- Explications

Il s'agit de la classe **Exception**. **ErrorException** est une autre classe qui hérite de celle-ci.

-

Question

Quel type d'erreur est lancé lorsqu'on n'intercepte pas une exception ?

- Bonne réponse

Une erreur fatale

- Explications

C'est une erreur fatale qui est levée, interrompant le script pour nous dire qu'une exception n'a pas été attrapée.

-

Question

Sur les trois paramètres demandés par le constructeur de la classe **Exception**, combien sont obligatoires ?

- Bonne réponse

aucun

- Explications

Les trois paramètres sont facultatifs. Cependant, je vous conseille toujours de fournir un message d'erreur à votre exception.

-

Question

Quelles sont les deux méthodes que je peux réécrire si j'hérite de la classe **Exception** ?

- Bonne réponse

`__construct()` et `__toString()`

- Explications

Seules les méthodes `__construct` et `__toString` ne sont pas finales, toutes les autres le sont.

-

Question

Quelle fonction permet d'enregistrer une autre fonction en callback qui sera appelée à chaque fois qu'une telle erreur est rencontrée ?

- Bonne réponse

`set_error_handler`

- Explications

Il s'agit de `set_error_handler`. Les deux autres n'existent pas.

-

Question

Si j'exécute ce code, qu'est-ce qui s'affichera à l'écran ?

Code : PHP - [Sélectionner](#)

```
<?php
function additionner ($a, $b)
{
    if (!is_numeric ($a) OR !is_numeric ($b))
        throw new Exception ('Les deux paramètres doivent être
des nombres');

    return $a + $b;
}

try // On va essayer d'effectuer les instructions situées dans
ce bloc
{
    echo additionner (12, 3);
    echo additionner ('azerty', 54);
    echo additionner (4, 8);
}

catch (Exception $e) // On va attraper les exceptions
"Exception" s'il y en a une qui est levée
{
    echo 'Exception lancée. Message : ' . $e->getMessage();
}
?>
```

○ Bonne réponse

15Exception lancée. Message : Les deux paramètres doivent être des nombres

• Explications

Le premier calcul s'est effectué comme il faut, **15** est donc affiché et la prochaine instruction contenue dans le bloc s'exécute. Ensuite, une exception est lancée, à l'écran s'affiche donc, à la suite, *Exception lancée. Message : Les deux paramètres doivent être des nombres*. À partir de ce moment-là, l'exécution du bloc est **interrompue** et la troisième instruction ne sera jamais exécutée. 😊



Note : 20 / 20

[<= Retour au cours](#)