

Les Exceptions PHP

Error Reporting

- Quand le programme est **complexe**. Avec de nombreux fichiers, classes, dépendances, etc.
- Écrit en **Orienté Objet**, standard mais personnalisable.
- Obtenir une **pile d'appel complète**.
- Quand il s'agit d'une erreur de programmation, qui sera ***"invisible"*** au parser de PHP et à l'exécution du code.

Instancier une **Exception**

- Quand le contexte logique l'impose, on va *"jeter"* (throw) une instance d'Exception (ou un objet héritant de l'Exception).

```
throw new Exception('Erreur quelconque', 123, $Throwable);
```

- (**string**)[**option**] : Un message d'erreur personnalisé, mais qui doit rester explicite.
- (**integer**)[**option**] : Une numéro de code erreur personnalisé, mais qui dépendra d'un convention locale ou de votre convention propre.
- (**throwable**)[**option**] : Un autre objet *"jetable"* (throwable), une exception généralement, pour pouvoir empiler les rapports.

Attraper une Exception

- Il n'y a plus qu'à "*attraper*" l'erreur au vol et l'afficher.

```
try {  
    // do something may send exceptions...  
}  
catch (Exception $error) {  
    echo $error . "\n"  
}
```

- **NB** : On peut faire **echo** directement sur l'objet Exception : celui utilise une fonction magique `__toString()` qui sert à sérialiser l'objet à la volée.

Personnaliser une Exception

- Il est possible de créer sa propre classe et de l'étendre avec l'objet Exception.
- Vous héritez ainsi des propriétés et méthodes de l'objet Exception.

```
class MonException extends \Exception { /*...*/ }
```

- Il n'y plus qu'à instancier et jeter votre Exception là où c'est nécessaire.

```
throw new MonException('Mon objet perso !');
```

Sources

- L'objet Exception PHP
 - <https://www.php.net/manual/fr/language.exceptions.php>
- Les Exceptions du SPL (Standard PHP Librairie)
 - <https://www.php.net/manual/fr/spl.exceptions.php>
- Les Exceptions étendues
 - <https://www.php.net/manual/fr/language.exceptions.extending.php>