

Construire un site dynamique, partie 2

Vous avez obtenu la note de 0 sur 6

Le 07/03/2018 à 09:44 vous aviez obtenu la note de 2 sur 6

Le 18/04/2018 à 19:34 vous aviez obtenu la note de 3 sur 6

À quoi sert **Composer** ?

Explication:

Composer est un gestionnaire de dépendances, il permet de les définir ainsi que leur version, de facilement les inclure dans un projet et de les mettre à jour selon des critères définis.

VÀ gérer les dépendances de mon projet et leurs versions

À organiser la structure de mon projet

À créer une partition musicale en PHP

À gérer les mises à jour de mon site

Je viens de récupérer un projet qui contient un `composer.json` et un `composer.lock`.

Je souhaite installer les dépendances, puis mettre à jour `acme/foo` selon le critère de version défini dans le `composer.json`.

Quelle(s) commande(s) dois-je exécuter ?

Explication:

Si les dépendances ne sont pas encore installées, c'est d'abord la commande `install` qu'il faut exécuter afin d'installer les dépendances aux versions spécifiquement décrites dans le `composer.lock`.

Une fois cette commande effectuée, il faut lancer `update acme/foo` afin de mettre à jour ce package uniquement, selon les critères définis dans le `composer.json`.

```
php composer install  
php composer update acme/foo
```

```
php composer update
```

```
php composer install  
php composer update
```

```
php composer install --update=acme/foo
```

J'ai déjà un fichier `composer.json` qui contient une liste de dépendances liées à mon projet.

Comment j'ajoute en ligne de commandes la librairie `acme/foo` en précisant que je ne veux que la version `1.0.x` la plus récente ?

Explication:

La commande `require` permet de mettre à jour le fichier `composer.json` avec la dépendance spécifiée `acme/foo` à la version `1.0.*`.

Ce qui signifie que la version la plus récente correspondant à `1.0.*` sera installée.

```
composer install acme/foo <2.0
```

```
composer require acme/foo:1.0.*
```

```
composer update acme/foo @ 1.0.*
```

```
composer include acme/foo 1.x
```

Que signifie la requête suivante :

```
SELECT id, color, price
FROM pens
WHERE color='ff0000'
LIMIT 15 OFFSET 10;
```

Explication:

La limite correspond au nombre de résultats retournés, l'offset, l'indice à partir duquel il faut sélectionner les lignes du rowset.

Afficher les crayons rouges situés du 10ème au 25ème emplacement

VAfficher les crayons rouges situés du 11ème au 26ème emplacement

Afficher tous les crayons situés du 10ème au 25ème emplacement

Afficher les crayons rouges situés du 16ème au 26ème emplacement

Comment obtenir l'offset en fonction du numéro de page (si on souhaite des pages de 20 enregistrements) ?

Explication:

Si `$page` vaut 1, on fait $(1-1)20 = 0$, soit un offset nul pour la première page. Dans le cas où `$page` vaut 2, $(2-1)20 = 20$, soit un offset de 20, et ainsi de suite.

```
$offset = (page * 20;
```

```
V$offset = ($page - 1) * 20;
```

```
$offset = ($page + 1) * 20;
```

Sur une page, vous trouvez un élément `<select>` proposant de choisir le nombre de résultats par page (10, 20 ou 30) :

Explication:

C'est le paramètre `LIMIT` qui changera, puisque par définition, il indique le nombre de résultats qu'on souhaite recevoir.

VCe chiffre va devenir le paramètre `LIMIT` de la requête SQL

Ce chiffre va devenir le paramètre `OFFSET` de la requête SQL

Ce chiffre, auquel on soustrait 1, sera utilisé lors de la requête SQL