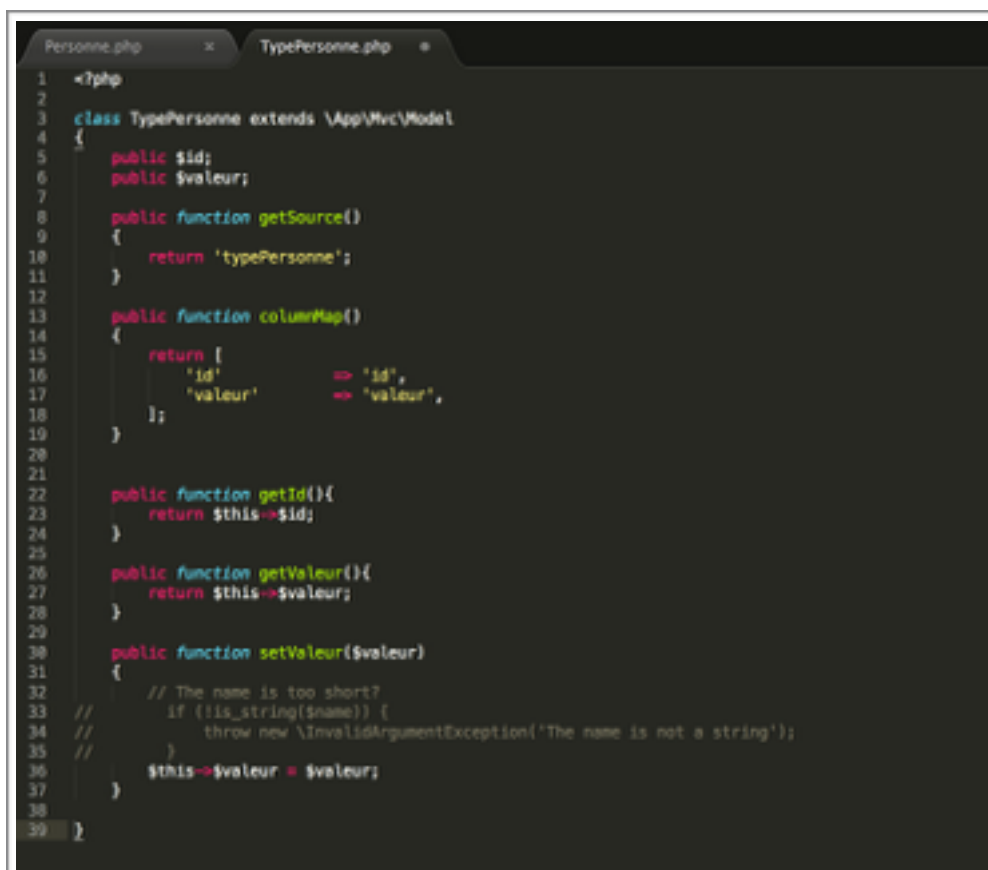


COMPTE RENDU DU START UP DAY N°5

Depuis la dernière séance, j'avais commencé à installer le framework Phalcon sur ma machine afin de récupérer le travail de Karine mais ce fut extrêmement compliqué. Aujourd'hui en fin de journée après je ne sais plus combien de manipulations à droite à gauche, j'ai pu installer Phalcon. Néanmoins la tâche s'avère un peu plus compliquée car il est facile de créer un projet mais très difficile de récupérer le projet de Karine.

Etant donné que Karine avait commencé à créer le modèle *Personne* et que c'était moi qui avait commencé à travailler sur le modèle de notre base de données, elle m'a demandé de continuer ses modèles tout en assurant l'affichage des données. Cette tâche s'avère facile mais un peu longue car il faut répéter les mêmes choses pour chaque modèle. Sachant que je n'ai toujours pas pu récupérer le projet de Karine car je n'ai pas réussi à configurer phalcon-devtool, je me suis basée sur son modèle *Personne* ainsi que des docs sur Internet pour créer les modèles restant.

Voici ci-dessous un exemple pour le modèle *TypePersonne*. Je l'ai fait pour tous nos modèles de données. J'ai commencé à coder la partie gestion de l'affichage selon l'attribut sélectionné dans le modèle. Cette tâche me demandera un peu plus de temps car il y a bien sûr des données que nous ne voulons pas afficher. Ces méthodes d'affichage nous permettront notamment de rechercher telle ou telle information dans notre application.



```
1 <?php
2
3 class TypePersonne extends \App\Mvc\Model
4 {
5     public $id;
6     public $valeur;
7
8     public function getSource()
9     {
10         return 'typePersonne';
11     }
12
13     public function columnMap()
14     {
15         return [
16             'id' => 'id',
17             'valeur' => 'valeur',
18         ];
19     }
20
21
22     public function getId(){
23         return $this->$id;
24     }
25
26     public function getValeur(){
27         return $this->$valeur;
28     }
29
30     public function setValeur($valeur)
31     {
32         // The name is too short?
33         // if (!is_string($name)) {
34         //     throw new \InvalidArgumentException('The name is not a string');
35         // }
36         $this->$valeur = $valeur;
37     }
38
39 }
```