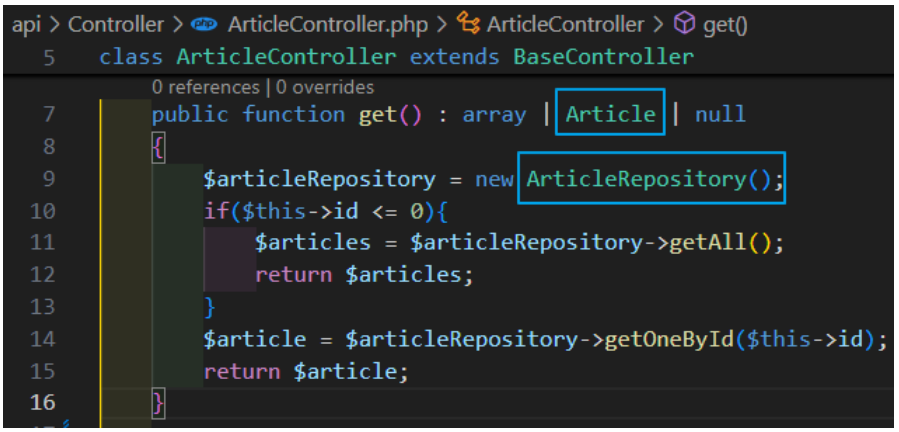
# PHP Blog Project

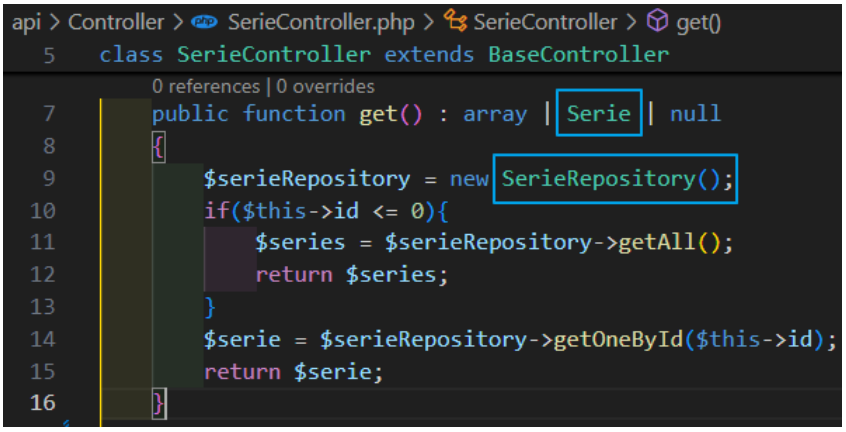
## 16. API Rest : Mise à jour des données (Update)

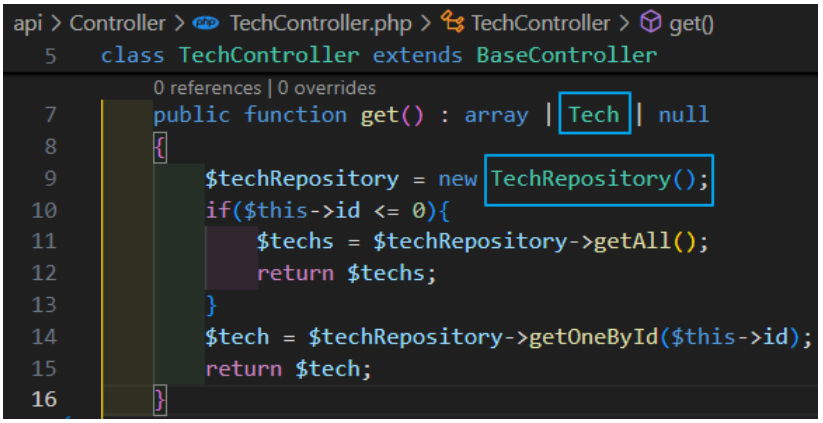
### Méthodes get() dans BaseController

Si nous comparons les méthodes get() déjà écrites dans les 3 contrôleurs nous constatons les différences suivantes

- Le type de retour (ligne 7)
- Le repository instancié (ligne 9)

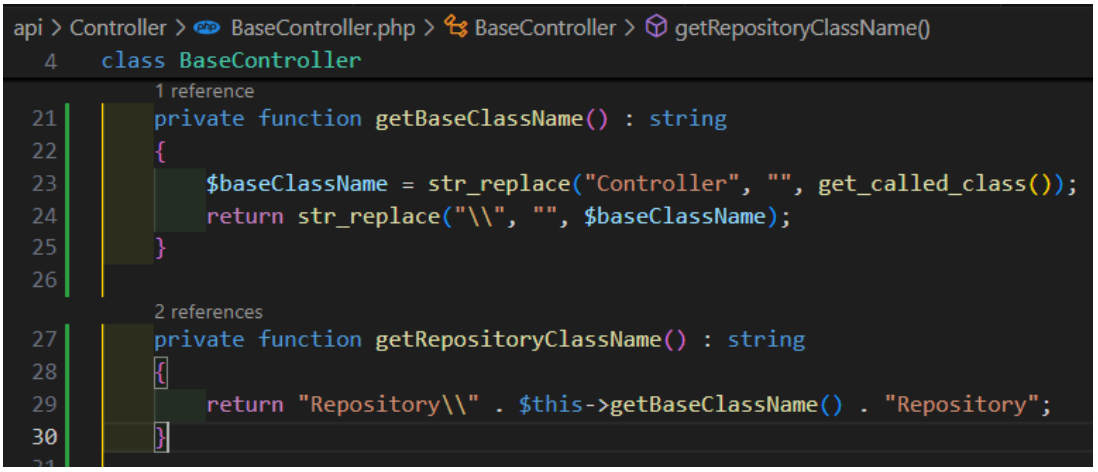


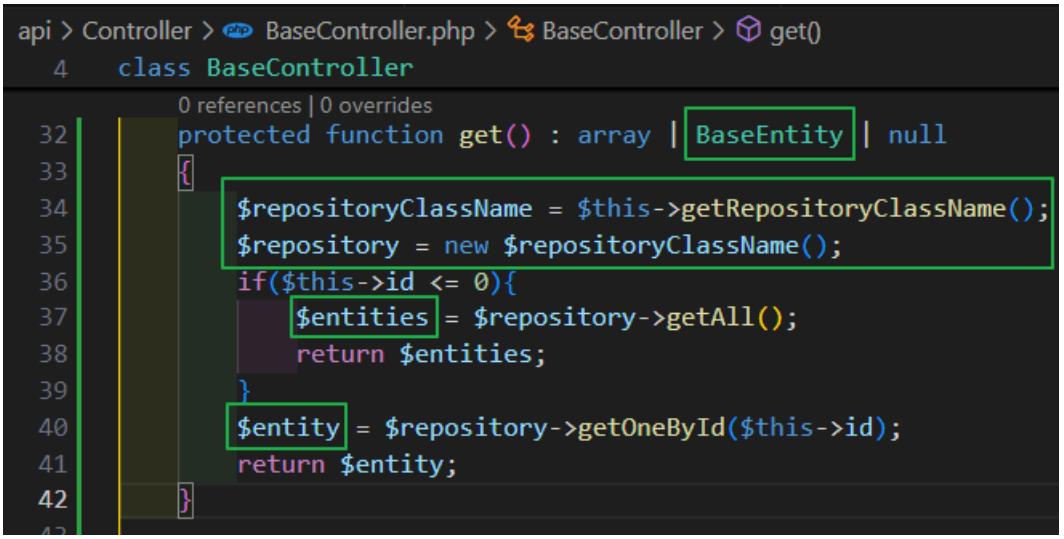




Pour le reste, seul le nom des variables qui stockent l'instance du Repository ($articleRepository, $serieRepository, $techRepository) et le nom des entités renvoyées par la méthode ($article, $serie, $tech) varient mais au final elles stockent la même chose.

Le type de retour peut être remplacé par BaseEntity qui est la classe mère d'Article, Serie et Tech
Pour remplacer la classe Repository, nous avons besoin d'un nom de classe dynamique. Nous pouvons nous inspirer de ce que nous avons déjà fait dans BaseRepository avec la méthode getEntityClassName()
Nous créons donc 2 méthodes qui vont nous permettre de récupérer le nom de la classe Repository dynamiquement.



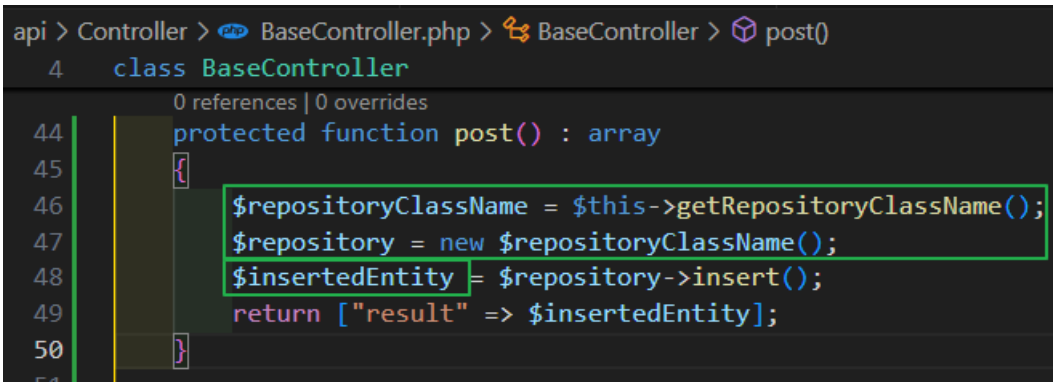

Nous pouvons maintenant écrire notre méthode get() dans BaseController à l'aide de la méthode getRepositoryClassName(). Nous modifions également les noms de variable spécialisés avec des termes plus génériques ($entities, $entity)




Il reste à commenter les méthodes get() dans les classes filles et tester avec Thunder Client

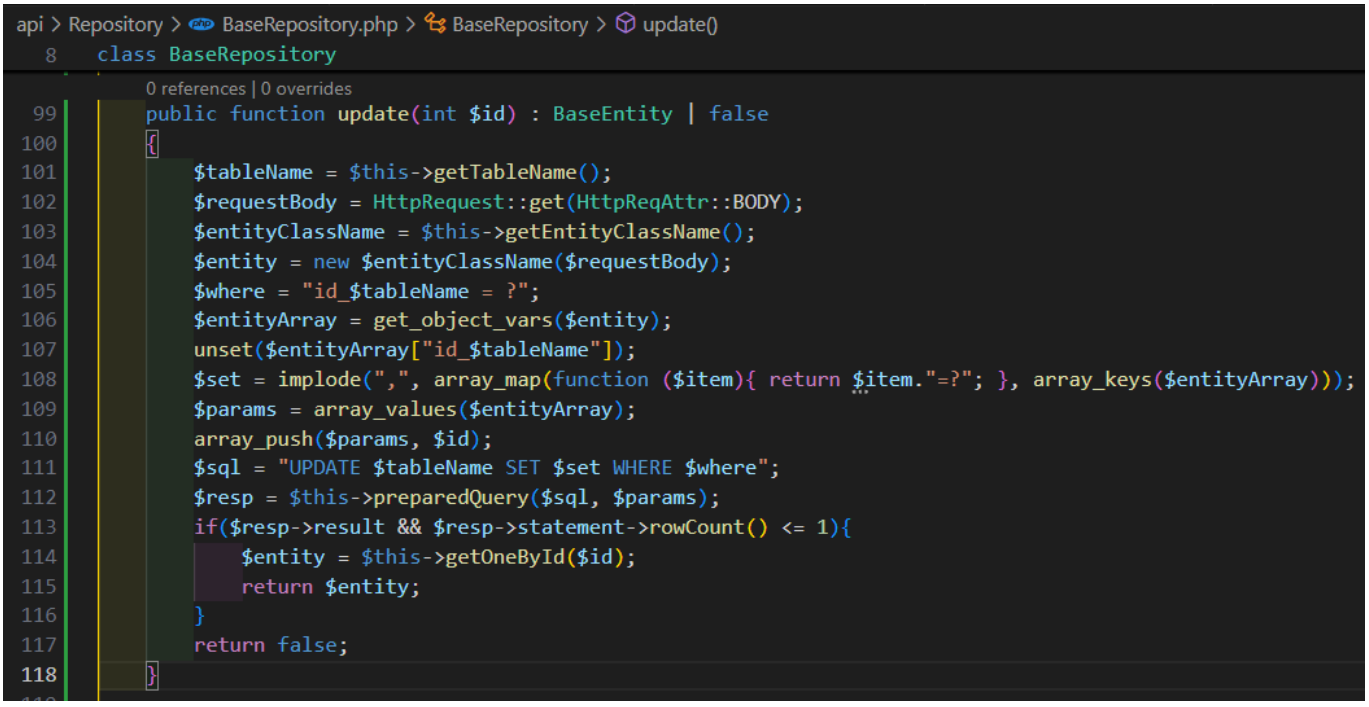
## Méthodes post() dans BaseController

Sur le même principe que pour les méthodes get() :




Commentez également les méthodes post() des classes filles et testez à Thunder Client.

## La méthode update de BaseRepository

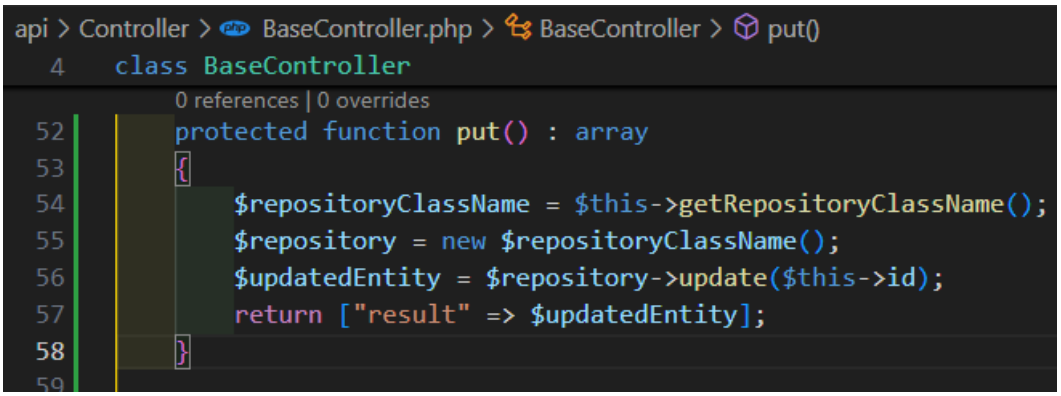



Les premières lignes de la méthode servent à construire les variables $tableNAme, $set, $where et $params dont nous aurons besoin pour exécuter la requête préparée (ligne 112)
Pour rappel, structure d'une requête SQL insert :

```
UPDATE $tableName SET $set WHERE $where
```

Si la requête s'est bien passé (ligne 113) nous renvoyons l'entité mise à jour, sinon nous renvoyons false.

## La méthode put de BaseControlleur




Très similaire à la méthode post()

A vous de **coder la suppression de données**, les méthodes delete dans le contrôleur et dans le repository (elles ont le même nom)

git :
https://github.com/DWWM-23526/PHP_BLOG_PROJECT/tree/Step16