

Challenge: Global Store



36KB

global-store-starter.zip
archive

Download



Open

Starter code has a missing form validation on the Date : @NOTNULL Annotation.

The code in this project is getting messy. In this challenge, you will refactor it using the 3 Layer architecture:

```
StoreController.java X
src > main > java > lu > cnfpcjavadeveloper > global_store > StoreController.java > Language Support for Java(TM) by Red Hat > StoreController > har
19 public class StoreController {
23     public String getForm(Model model, @RequestParam(required = false) String id) {
27     }
28
29     @PostMapping("/submitItem")
30     public String handleSubmit(@Valid Item item, BindingResult result, RedirectAttributes redirectAttributes) {
31         if (item.getPrice() < item.getDiscount()) {
32             result.rejectValue("price", "", "Price cannot be less than discount");
33         }
34         if (result.hasErrors())
35             return "form";
36         int index = getIndexFromId(item.getId());
37         String status = Constants.SUCCESS_STATUS;
38         if (index == Constants.NOT_FOUND) {
39             items.add(item);
40         } else if (within5Days(item.getDate(), items.get(index).getDate())) {
41             items.set(index, item);
42         } else {
43             status = Constants.FAILED_STATUS;
44         }
45         redirectAttributes.addFlashAttribute("status", status);
46         return "redirect:/inventory";
47     }
```

Step 1

Organize the data store and all CRUD operations within the **Repository Layer**.

Step 2

Move all business logic and decision-making processes to the **Service Layer**.

This layer should serve as an intermediary, interacting with the Repository to execute business operations.

Step 3

Ensure the **Controller Layer** is solely responsible for managing the model and rendering the view.

Step 4

Test your application to verify that everything functions correctly after the changes.