

Peer Review of Group 101

By Group 12

S1800918

S1813305

Implementation:

APricingPolicy is a really vague name for a policy. Could be more descriptive like DefaultPricingPolicy.

This is because if there are other pricing policies implemented in the future, it will be hard to differentiate one from the other from first glance.

Could throw exception that the length of the duration of discount > 0.

removeDiscount is a bad method because we potentially remove a discount in between 2 discount ranges

If Provider holds a reference to a PricingPolicy then removeDiscount function will never be able to be called after it was created as it is not in the interface

(exception is casting but doing if typeof(..) then you cast is a bad coding practice)

Good use of JavaDocs, and JavaML.

Tests:

Some tests (eg. negativeDateRange), could be tested by unit tests in DateRange itself to avoid repetitive testing and also perhaps have something in DateRange that handles that situation.

Maybe add test in front of the test function names to make it clearer that they are tests.

multipleDiscounts is a really good test because it covers several test cases at the same time.

Conclusion:

Overall, we think this is a rather well written code. This code could be refactored better for maintainability.