Some notes for the code reviewer:

* Please find the UML diagram **Basket\_UML.jpg** to understand the class relationships.
* I have preferred ‘Packages by feature’ over ‘Packages by layer’ for package structure.
* Basket has been priced using ProductPricingService where discount has been applied to the unit cost of product.
* The server (DAO) class notifies users by throwing exception if product unit price is not assigned to the product.
* Automated tests have been implemented for BasketService class which indirectly tests methods in other services and DAOs.
* Last but very important, this is the maven project and can simply be run from command line using a command ‘mvn clean install’ – the automated test method **printBasket(Basket)** in **BasketServiceTest.java** will print the output of basket (i.e. all basket items, its unit cost, discounted price and basket total cost). Try changing the product information in the test class and re-run the test, this will give you correct expected output in following example format,

===========BASKET INFORMATION STARTS===========

Product: Banana, Price: 1.5, Quantity: 5, Cost: 7.5, Discount: 0.01, Discounted Price: 7.425

Product: Orange, Price: 1.75, Quantity: 3, Cost: 5.25, Discount: 0.0, Discounted Price: 5.25

Product: Peach, Price: 2.5, Quantity: 2, Cost: 5.0, Discount: 0.02, Discounted Price: 4.9

Product: Apple, Price: 2.0, Quantity: 6, Cost: 12.0, Discount: 0.05, Discounted Price: 11.4

Product: Lemon, Price: 0.5, Quantity: 10, Cost: 5.0, Discount: 0.0, Discounted Price: 5.0

BASKET TOTAL COST: 33.975

===========BASKET INFORMATION ENDS===========

While I appreciate there may be chances of improvements but based on some presumptions and keeping time constraints in mind I have tried my level best to develop the solution to the best of my knowledge. Hope you like my solutions and given an opportunity I would like to discuss area of improvements in more details.

Happy Coding!!