

1 Create the following application (USE A TDD based approach)

2

3 Create the maven application and add the relevant dependencies discussed during theory session

4

5 Class Product(int productId, String productName,String productDescription, double price,int quantity)

6 Provide the appropriate Constructors,accessors,mutators,toString

7

8 Create a properties file/ Helper class as discussed during theory session for declaring and loading jdbc related configuration

9

10 Interface ProductDAO

11 void createConnection()

12 void closeConnection()

13 void addProduct(Product product)

14 Product getProduct(int productId)

15 List<Product>getAllProducts()

16 void updateProduct(Product product)

17 void deleteProduct(int productId)

18 Class ProductDAOImpl that implements the ProductDAO and appropriately provides the implementation to the

19 functionality.

20

21 Class ProductSortingUtility with below methods that sort and print the products accordingly

22 void sortDefault(List<Product> products) (by productId)

23 void sortByProductName(List<Product> products)

24 void sortByProductPrice(List<Product> products)

25 void sortByProductQuantity((List<Product> products))

26

27 Interface ProductService

28 void createProduct(Product product)

29 void findProductById(int productId)

30 List<Product> findAllProducts()

31 void modifyProduct(Product product)

32 void removeProduct(int productId)

33 void sortById(List<Product> products) (by productId)

34 void sortByProductName(List<Product> products)

35 void sortByProductPrice(List<Product> products)

36 void sortByProductQuantity((List<Product> products))

37

38 *** Write Down the unit tests for services / integrate the jacoco coverage and sonarqube analysis for the Code

39

40 Class ProductServiceImpl that implements the ProductService and appropriately implements the methods

41 This service will use the ProductSortingUtility for relevant sorting operations

42

43

44 **Class TestProductSorting**

45 **main method.**

46

47 **Add about 10 products to table**

48 **Show a menu for listing the products as per sorting criteria.**

49 **Test the other CRUD methods as exposed by service.**

50