Spring JDBC-Template Hands On Billing Software Application

Important Instructions:

- Please read the document thoroughly before you code.

- Import the given skeleton code into your Eclipse.

- Do not change the Skeleton code or the package structure, method names, variable names, return types, exception clauses, access specifiers etc.

- You can create any number of private methods inside the given class.

- You can test your code by running main() method of the program

- Using **Spring Maven** to develop the application

General Design Constraints:

- Ensure that all the Java Coding Standards are followed.

- Assume that the method inputs are valid always, hence exceptional blocks are not needed to be included in the development.

Problem Statement:

A TakeHome company requires billing software for its super market where different types of products are sold to the customers on retail basis.

This billing software should help

a. The cashier to see the product information by entering the product code

b. Calculate the total cost for each product based on the quantity

c. Display the final bill to user

d. Throw exception for invalid details

Skeleton File for Development:

Import the below attached skeleton code into your eclipse project and implement the required functionalities

Technical Requirements:

Requirement 1:Database design

Use createScripts.sql file for the database requirement

Spring JDBC-Template Hands On Billing Software Application

Requirement 2:Bean class

Create a Product bean class(com.cts.bean) with following members.

Variable Name	Data type
product_code	Int
product_name	String
product_category	String
product_description	String
product_price	double

Requirement 3:DAOInterface
Write a ProductDao interface (com.cts.dao) with following members.

Interface Name	Method Name	Input Parameters	Output Parameters
ProductDao	getProductDetails();	Int product_code	Product

Requirement 4:DAO class implantation
Write a ProductDaoImpl class (com.cts.dao) which implements
ProductDao interface with following members.

Class Name	Method Name	Input Parameters	Output Parameters	Logic
ProductDaoImpl	getProductDetails	Int product_code	Product	Query to DB select a product by product_code using jdbcTemplte.queryForObject() method and return the object.

Requirement 5:Service class Impl
Write a ProductService class (com.cts.service) which has following members.

Class Name	Method Name	Input Parameters	Output Parameters	Logic

		Parameters	Parameters	
ProductService	getProductDetails	Int product_code	Product	Invoke the ProductDao

Spring JDBC-Template Hands On Billing Software Application

				getProductDetails() return the object.
ProductService	calculatePrice()	Int quantity,double product_price	double	Use quantity and price calculate the total amount and return the value.
ProductService	validateProductCode()	Int product_code	boolean	Check product_code should be >0 and 4 digit number return true else return false
ProductService	validateQuantity()	int quantity	boolean	Check quantity should be >0 return true else return false

Requirement 6:Exception Class

Write a BillingException class (com.cts.service) which extends Exception classraise
the exception as per the requirement.

Requirement 7:MainClass

In main method class to test your application. Use switch case to perform as per
requirement.

Note: Throw exception called **BillingException** with appropriate error message for the
following cases and get input until user enters the valid information
- Quantity should not be lesser than or equal to zero otherwise raise the exception
- Product code should be 4 digit positive number otherwise raise the exception

Spring JDBC-Template Hands On Billing Software Application

SAMPLE INPUT and OUTPUT

****
        1. Generate Bill by entering code and quantity
        2. Exit
        1
        Enter product_code
        Invalid product_code, it should be >0 and 4 digit +ve number
        896
        Invalid product_code, it should be >0 and 4 digit +ve number
        Enter product_code
        -75
        Invalid product_code, it should be >0 and 4 digit +ve number
        Enter product_code
        0
        Invalid product_code, it should be >0 and 4 digit +ve number
        Enter product_code
        1002