package com.nttdata.petstore.driver;

import java.sql.Connection;

import java.util.ArrayList;

import java.util.Iterator;

import java.util.List;

import java.util.Scanner;

import org.apache.log4j.Logger;

import com.nttdata.petstore.dao.PetStoreDAOException;

import com.nttdata.petstore.dao.ProductDAO;

import com.nttdata.petstore.dbcon.ConnectionHolder;

import com.nttdata.petstore.dbcon.DBConnectionException;

import com.nttdata.petstore.dbfw.DBFWException;

import com.nttdata.petstore.domain.Cart;

import com.nttdata.petstore.domain.CartItem;

import com.nttdata.petstore.domain.Customer;

import com.nttdata.petstore.domain.Item;

import com.nttdata.petstore.service.PetStoreException;

import com.nttdata.petstore.service.PetStrorefacade;

public class PetStoreDriver {

final static Logger LOG = Logger.getLogger("ProductDAO.class");

/\*\*

\* @param args

\* @throws PetStoreException

\* @throws DBFWException

\* @throws PetStoreDAOException

\*/

public static void main(String[] args) throws PetStoreException,

DBFWException, PetStoreDAOException {

// TODO Auto-generated method stub

Connection con = null;

ConnectionHolder holder=null;

try {

holder=ConnectionHolder.getInstance();

con=holder.getConnection();

} catch (DBConnectionException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

if (con != null) {

LOG.debug("checking connection");

System.out.println("The Connection is Established");

}

PetStrorefacade facade = new PetStrorefacade();

Customer customer = new Customer();

int chooseOption;

int categId = 0;

int prodId = 0;

int itemId = 0;

List categoryList = new ArrayList();

List specificCategoryList = new ArrayList();

List productList = new ArrayList();

List specificProductList = new ArrayList();

List itemList = new ArrayList();

List specificItemList = new ArrayList();

Cart shoppingCart = new Cart();

CartItem cartItem = new CartItem();

Item item = new Item();

String userId = null;

String password = null;

String firstName = null;

String lastName = null;

String DOB = null;

String address = null;

int pNumber = 0;

int cNumber = 0;

String cType = null;

String eDate;

int quantity = 0;

int order = 0;

boolean isTrue = false;

boolean isRegistered = false;

Scanner scan = new Scanner(System.in);

System.out

.println("Enter your choice 1.Registered 2.New User:");

int choice = scan.nextInt();

switch (choice) {

case 1: {

LOG.info("taking input from user");

LOG.info("taking input from user");

System.out.println("Enter the userID");

userId = scan.next();

System.out.println("Enter the password");

password = scan.next();

isTrue = facade.validateUser(userId, password);

if (isTrue) {

LOG.debug("validating user");

System.out.println("Registered user");

break;

} else {

System.out.println("Not a Registered user");

}

break;

}

case 2: {

if (isTrue == false) {

LOG.debug("validating user Registration");

System.out.println("Please Register");

} else {

System.out.println(" New User Registration");

}

LOG.info("New user registration");

System.out.println("Enter the User Id");

userId = scan.next();

System.out.println("Enter Password ");

password = scan.next();

System.out.println("Enter Your First Name ");

firstName = scan.next();

System.out.println("Enter your Last Name ");

lastName = scan.next();

System.out

.println("Enter your date of Bith in the Format DD-MMM-YY ");

DOB = scan.next();

System.out.println("Enter the Address ");

address = scan.next();

System.out.println("Enter the phone number ");

pNumber = scan.nextInt();

System.out.println("Enter the Credit Card number ");

cNumber = scan.nextInt();

System.out.println("Enter the Credit Card Type ");

cType = scan.next();

System.out

.println("Enter the Credit Card expiry Date in the Format DD-MMM-YY ");

eDate = scan.next();

customer.setCustId(userId);

customer.setPasword(password);

customer.setFirstName(firstName);

customer.setLastName(lastName);

customer.setDateOfBirth(DOB);

customer.setAddress(address);

customer.setContactNumber(pNumber);

customer.setCreditCardno(cNumber);

customer.setCardType(cType);

customer.setCardExpiryDate(eDate);

facade.registerUser(customer);

if (isRegistered) {

LOG.debug("checking the registration details");

System.out.println("Registered");

} else {

System.out.println(" Please enter valid details");

//throw new PetStoreException(

// "Valid entries not entered", null);

}

}

}

System.out.println("Catagories : ");

categoryList = facade.getCategories();

System.out.println(categoryList);

System.out.println("Choose the Category ");

categId = scan.nextInt();

specificCategoryList = facade.getCatByid(categId);

System.out.println("The Category List:");

System.out.println(specificCategoryList);

productList = facade.getProductList(categId);

System.out.println(productList);

System.out

.println("Enter the productID of the product desired");

prodId = scan.nextInt();

specificProductList = facade.getProduct(categId, prodId);

System.out.println(specificProductList);

System.out.println("The Item List is ");

itemList = facade.getItemList(categId, prodId);

System.out.println(itemList);

System.out.println("Enter the ItemID for the given product");

itemId = scan.nextInt();

System.out.println("The specific item list is");

try {

specificItemList = facade.getItem(categId, prodId, itemId);

} catch (DBConnectionException e) {

// TODO Auto-generated catch block

//e.printStackTrace();

throw new PetStoreDAOException(e.getMessage());

}

System.out.println(specificItemList);

Iterator iterate = specificItemList.iterator();

while (iterate.hasNext()) {

LOG.debug("iterating through the Item");

item = (Item) iterate.next();

System.out.println(item);

}

LOG.info("Entering the sale details");

System.out.println("Enter the quantity");

quantity = scan.nextInt();

order = facade.getOrderId();

shoppingCart.setOrderId(order);

shoppingCart.setCustId(userId);

shoppingCart.addCartItem(item, quantity);

System.out.println(facade.insertNewOrder(shoppingCart));

System.out.println("enter the order id");

int ord = scan.nextInt();

facade.getPurchaseDetails(ord);

}

}