Main Method – Controller

package com.sssit.product.controller;

import java.util.List;

import java.util.Scanner;

import com.sssit.product.pojo.ProductPojo;

import com.sssit.product.repository.ProductRepository;

public class ProductManagement {

static Scanner ip = new Scanner(System.in);

static ProductRepository productRepository = new ProductRepository();

public static void main(String[] args) {

char ch =' ';

do

{

System.out.println("1. Insert New Product");

System.out.println("2. Update MRP by Id");

System.out.println("3. Remove Product By Id");

System.out.println("4. Retreive all Products");

System.out.println("5. Retrieve Product By Id");

System.out.println("6. Exit");

System.out.println("Enter your choice:");

int option = ip.nextInt();

switch(option) {

case 1:

addNewProduct();

break;

case 2:

updateMrpById();

break;

case 3:

removeProductById();

break;

case 4:

retrieveAllProducts();

break;

case 5:

retrieveProductById();

break;

default:

return;

}

System.out.println("Do you want to continue....");

ch = ip.next().charAt(0);

}while(ch=='y');

}

public static void addNewProduct() {

System.out.println("Enter Product Id?");

int pid = ip.nextInt();

System.out.println("Enter Product Name?");

String pname = ip.next();

System.out.println("Enter Product MRP?");

double mrp = ip.nextDouble();

int result = productRepository.insertProduct(pid, pname, mrp);

if(result==0)

System.out.println("Failed to Insert the data");

else

System.out.println("Inserted Successfully....");

}

public static void updateMrpById() {

System.out.println("Enter Product Id?");

int pid = ip.nextInt();

List<ProductPojo> prods =

productRepository.retrieveProductById(pid);

if(prods.isEmpty()) {

System.out.println("Product Id is invalid");

return;

} else {

for(ProductPojo prod : prods) {

System.out.println("Product Id =" + prod.getPid());

System.out.println("Product Name = " + prod.getpName());

System.out.println("Product MRP = " + prod.getMrp());

}

System.out.println("Do you want to change MRP?");

char opt = ip.next().charAt(0);

if(opt=='y') {

System.out.println("Enter Product MRP?");

double mrp = ip.nextDouble();

int result = productRepository.updateMrpById(pid, mrp);

if(result==0)

System.out.println("Failed to Update MRP");

else

System.out.println("MRP Upated Successfully....");

}

}

}

public static void removeProductById() {

System.out.println("Enter Product Id?");

int pid = ip.nextInt();

List<ProductPojo> prods =

productRepository.retrieveProductById(pid);

if(prods.isEmpty()) {

System.out.println("Product Id is invalid");

return;

} else {

for(ProductPojo prod : prods) {

System.out.println("Product Id =" + prod.getPid());

System.out.println("Product Name = " + prod.getpName());

System.out.println("Product MRP = " + prod.getMrp());

}

System.out.println("Do you want remove the Product?");

char opt = ip.next().charAt(0);

if(opt=='y') {

int result = productRepository.removeProductById(pid);

if(result==0)

System.out.println("Failed to Delete Product");

else

System.out.println("Product Info Deleted Successfully....");

}

}

}

public static void retrieveAllProducts() {

List<ProductPojo> prods =

productRepository.retrieveAllProducts();

if(prods.isEmpty()) {

System.out.println("Product Table is Empty");

return;

} else {

for(ProductPojo prod : prods) {

System.out.printf("%10d%20s%10f\n", prod.getPid(),prod.getpName(),prod.getMrp());

}

}

}

public static void retrieveProductById() {

System.out.println("Enter Product Id?");

int pid = ip.nextInt();

List<ProductPojo> prods =

productRepository.retrieveProductById(pid);

if(prods.isEmpty()) {

System.out.println("Product Id is invalid");

return;

} else {

for(ProductPojo prod : prods) {

System.out.println("Product Id =" + prod.getPid());

System.out.println("Product Name = " + prod.getpName());

System.out.println("Product MRP = " + prod.getMrp());

}

}

}

}

Repository

package com.sssit.product.repository;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.util.ArrayList;

import java.util.List;

import com.sssit.product.pojo.ProductPojo;

import com.sssit.product.util.JDBCConnection;

public class ProductRepository {

Connection con = null;

PreparedStatement pstmt = null;

public int insertProduct(int pid,String pname,double mrp) {

try

{

con=JDBCConnection.getConnection();

final String SQLQuery = "insert into product values(?,?,?)";

pstmt = con.prepareStatement(SQLQuery);

pstmt.setInt(1, pid);

pstmt.setString(2, pname);

pstmt.setDouble(3,mrp);

return pstmt.executeUpdate();

}

catch (Exception e) {

// TODO: handle exception

}

return 0;

}

public int updateMrpById(int pid,double mrp) {

try

{

con=JDBCConnection.getConnection();

final String SQLQuery = "update product set mrp=? where pid=?";

pstmt = con.prepareStatement(SQLQuery);

pstmt.setInt(2, pid);

pstmt.setDouble(1,mrp);

return pstmt.executeUpdate();

}

catch (Exception e) {

// TODO: handle exception

}

return 0;

}

public List<ProductPojo> retrieveProductById(int pid) {

List<ProductPojo> productsList = new ArrayList<ProductPojo>();

try

{

con=JDBCConnection.getConnection();

final String SQLQuery = "select \* from product where pid=?";

pstmt = con.prepareStatement(SQLQuery);

pstmt.setInt(1, pid);

ResultSet rs = pstmt.executeQuery();

while(rs.next()) {

ProductPojo product = new ProductPojo();

product.setPid(rs.getInt("pid"));

product.setpName(rs.getString("pname"));

product.setMrp(rs.getDouble("mrp"));

productsList.add(product);

}

}

catch (Exception e) {

// TODO: handle exception

}

return productsList;

}

public int removeProductById(int pid) {

try

{

con=JDBCConnection.getConnection();

final String SQLQuery = "delete from product where pid=?";

pstmt = con.prepareStatement(SQLQuery);

pstmt.setInt(1, pid);

return pstmt.executeUpdate();

}

catch (Exception e) {

// TODO: handle exception

}

return 0;

}

public List<ProductPojo> retrieveAllProducts() {

List<ProductPojo> productsList = new ArrayList<ProductPojo>();

try

{

con=JDBCConnection.getConnection();

final String SQLQuery = "select \* from product";

pstmt = con.prepareStatement(SQLQuery);

ResultSet rs = pstmt.executeQuery();

while(rs.next()) {

ProductPojo product = new ProductPojo();

product.setPid(rs.getInt("pid"));

product.setpName(rs.getString("pname"));

product.setMrp(rs.getDouble("mrp"));

productsList.add(product);

}

}

catch (Exception e) {

// TODO: handle exception

}

return productsList;

}

}

POJO Class

**package** com.sssit.product.pojo;

**public** **class** ProductPojo {

**private** **int** pid;

**private** String pName;

**private** Double mrp;

**public** **int** getPid() {

**return** pid;

}

**public** **void** setPid(**int** pid) {

**this**.pid = pid;

}

**public** String getpName() {

**return** pName;

}

**public** **void** setpName(String pName) {

**this**.pName = pName;

}

**public** Double getMrp() {

**return** mrp;

}

**public** **void** setMrp(Double mrp) {

**this**.mrp = mrp;

}

@Override

**public** String toString() {

**return** "ProductPojo [pid=" + pid + ", pName=" + pName + ", mrp=" + mrp + "]";

}

}

Util Class to get the connection

package com.sssit.product.util;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

public class JDBCConnection {

private static Connection con = null;

public static final String URL = "jdbc:oracle:thin:@localhost:1521:orcl";

public static final String UNAME = "advdec23";

public static final String PASS = "advdec23";

public static Connection getConnection() {

if(con==null) {

try {

Class.forName("oracle.jdbc.driver.OracleDriver");

con = DriverManager.getConnection(URL, UNAME, PASS);

} catch (ClassNotFoundException e) {

System.out.println("Failed to Load Driver class...");

} catch (SQLException e) {

System.out.println("Failed to Establish connection...");

}

}

return con;

}

}