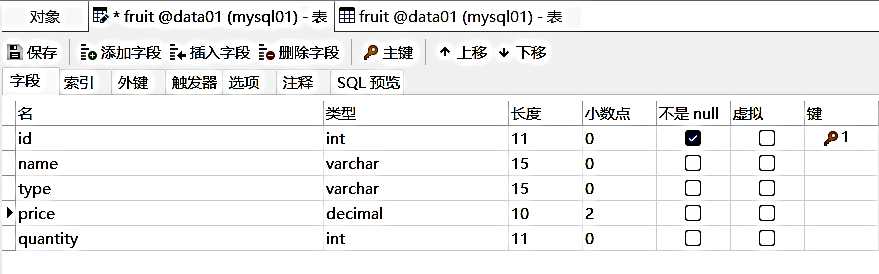
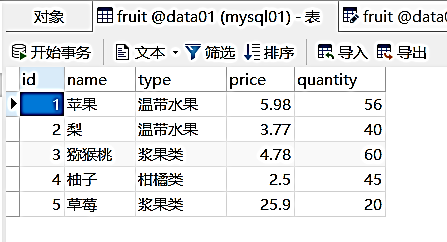
# 鑫鑫水果库存管理系统

## 数据库

### （一）数据库设计



### （二）fruit表



## 二、源代码

### （一）Fruit

package fruitManager;

import java.math.BigDecimal;

public class Fruit {

private int id;

private String name;

private String type;

private BigDecimal price;

private int quantity;

public Fruit() {

super();

}

public Fruit(int id, String name, String type, BigDecimal price, int quantity) {

super();

this.id = id;

this.name = name;

this.type = type;

this.price = price;

this.quantity = quantity;

}

public Fruit(int id) {

super();

this.id = id;

}

public Fruit(int id, int quantity) {

super();

this.id = id;

this.quantity = quantity;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getType() {

return type;

}

public void setType(String type) {

this.type = type;

}

public BigDecimal getPrice() {

return price;

}

public void setPrice(BigDecimal price) {

this.price = price;

}

public int getQuantity() {

return quantity;

}

public void setQuantity(int quantity) {

this.quantity = quantity;

}

}

### （二）FruitDao

package fruitManager;

public interface FruitDao {

boolean insert(Fruit fruit);

boolean delete(Fruit fruit);

boolean update(Fruit fruit);

Fruit select(String name);

static boolean fruitExists(String name) {

// TODO 自动生成的方法存根

return false;

}

}

### （三）JdbcUtils

package fruitManager;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

public class JdbcUtils {

public static final String className = "com.mysql.jdbc.Driver";

public static final String url = "jdbc:mysql://localhost:3306/data01";

public static final String user = "root";

public static final String password = "root";

public JdbcUtils() {

try {//加载数据库

Class.forName(className);

} catch (ClassNotFoundException e) {

// TODO 自动生成的 catch 块

e.printStackTrace();

}

}

public static Connection getConnection() {

try {//连接数据库

return DriverManager.getConnection(url, user, password);

} catch (SQLException e) {

// TODO 自动生成的 catch 块

e.printStackTrace();

}

return null;

}

public static void close(ResultSet res,PreparedStatement statement,Connection con) {

try {

if(res != null){

res.close();

}

if(statement != null) {

statement.close();

}

if(con != null) {

con.close();

}

} catch (SQLException e) {

// TODO 自动生成的 catch 块

e.printStackTrace();

}

}

}

### （四）FruitDao\_imp

package fruitManager;

import java.math.BigDecimal;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

public class FruitDao\_imp {

public static final String SQL\_INSERT=" INSERT INTO fruit VALUES(?,?,?,?,?)";

public static final String SQL\_DELETE=" DELETE FROM fruit WHERE id=?";

public static final String SQL\_UPDATE=" UPDATE fruit SET quantity=? WHERE id=?";

public static final String SQL\_SELECT=" SELECT \* FROM fruit WHERE name=?";

//1,添加

public boolean insert(Fruit fruit) {

Connection con = JdbcUtils.getConnection();

PreparedStatement statement = null;

try {

statement = con.prepareStatement(SQL\_INSERT);

statement.setInt(1,fruit.getId());

statement.setString(2,fruit.getName());

statement.setString(3,fruit.getType());

statement.setBigDecimal(4,fruit.getPrice());

statement.setInt(5,fruit.getQuantity());

int line = statement.executeUpdate();

return line>0?true:false;

} catch (SQLException e) {

// TODO 自动生成的 catch 块

e.printStackTrace();

}finally{

JdbcUtils.close(null, statement,con);

}

return false;

}

//2,删除(根据编号)

public boolean delete(Fruit fruit) {

Connection con = JdbcUtils.getConnection();

PreparedStatement statement = null;

try {

statement = con.prepareStatement(SQL\_DELETE);

statement.setInt(1, fruit.getId());

int line = statement.executeUpdate();

return line>0?true:false;

} catch (SQLException e) {

// TODO 自动生成的 catch 块

e.printStackTrace();

}finally {

JdbcUtils.close(null, statement, con);

}

return false;

}

//3,修改（根据编号修改库存数量）

public boolean update(Fruit fruit) {

Connection con = JdbcUtils.getConnection();

PreparedStatement statement = null;

try {

statement = con.prepareStatement(SQL\_UPDATE);

statement.setInt(1,fruit.getQuantity());

statement.setInt(2,fruit.getId());

int line = statement.executeUpdate();

return line>0?true:false;

} catch (SQLException e) {

// TODO 自动生成的 catch 块

e.printStackTrace();

}finally {

JdbcUtils.close( null, statement,con);

}

return false;

}

//4,查询(根据水果名称)

public Fruit select(String name) {

Connection con = JdbcUtils.getConnection();

PreparedStatement statement = null;

ResultSet res = null;

try {

statement = con.prepareStatement(SQL\_SELECT);

statement.setString(1,name);

res = statement.executeQuery();

while(res.next()) {

int id = res.getInt(1);

String name1 = res.getString(2);

String type = res.getNString(3);

BigDecimal price = res.getBigDecimal(4);

int quantity = res.getInt(5);

return new Fruit(id,name1,type,price,quantity);

}

} catch (SQLException e) {

// TODO 自动生成的 catch 块

e.printStackTrace();

}finally {

JdbcUtils.close(res,statement,con);

}

return null;

}

}

### （五）View

package fruitManager;

import java.math.BigDecimal;

import java.util.Scanner;

public class View {//水果库存管理

private static Scanner sc = new Scanner(System.in);

@SuppressWarnings("resource")

public static int MenuView() {

System.out.println("!——————-欢迎进入鑫鑫水果库存管理系统————————！");

System.out.println("!——————————请根据 提 示 进 行 操作——————————!");

Scanner scanner = new Scanner(System.in);

int option = -1;

while (option < 0 || option > 4) {

System.out.println("!——————————请根据 提 示 进 行 操作——————————!");

System.out.println("!——————————1——添加库存水果信息——————————!");

System.out.println("!——————————2——删除库存水果信息——————————!");

System.out.println("!——————————3——修改库存水果信息 ——————————!");

System.out.println("!——————————4——查询库存水果信息——————————!");

System.out.println("!——————————0——退 出——————————!");

System.out.println("!——————————————————————————————————!");

System.out.print("请输入您的选择（0-4）：");

String input = scanner.nextLine();

try {

option = Integer.parseInt(input);

if (option < 0 || option > 4) {

System.out.println("无效的选项，请输入一个介于 0 到 4 之间的数字。");

}

} catch (NumberFormatException e) {

System.out.println("输入无效，请输入一个数字。");

}

}

return option;

}

//添加

public static Fruit insertMenuView() {

System.out.println("请根据提示操作：");

System.out.println("请输入要添加的水果编号：");

int id = sc.nextInt();

sc.nextLine(); // 消耗掉nextInt()留下的换行符

System.out.println("请输入要添加的水果名称：");

String name = sc.nextLine();

System.out.println("请输入要添加的水果种类：");

String type = sc.nextLine();

System.out.println("请输入要添加的水果单价：");

BigDecimal price = sc.nextBigDecimal();

System.out.println("请输入要添加的水果数量：");

int quantity = sc.nextInt();

return new Fruit(id,name,type,price,quantity);

}

//1.删除（根据水果编号）

public static Fruit deleteMenuView() {

System.out.println("请根据提示操作：");

System.out.println("请输入要删除的水果编号：");

int deleteId = sc.nextInt();

sc.nextLine(); // 消耗掉nextInt()留下的换行符

return new Fruit(deleteId);

}

//2.修改（根据水果编号修改库存）

public static Fruit updateMenuView() {

System.out.println("请根据提示操作：");

System.out.println("请输入要修改的水果编号：");

int updateId = sc.nextInt();

sc.nextLine(); // 消耗掉nextInt()留下的换行符

System.out.println("请输入该编号的水果库存为：");

int updateQuantity = sc.nextInt();

sc.nextLine(); // 消耗掉nextInt()留下的换行符

return new Fruit(updateId,updateQuantity);

}

//3.查询（根据水果名称）

public static String SelectMenuView() {

System.out.println("请根据提示操作：");

System.out.println("请输入要查询的水果名字:");

String name = sc.nextLine();

return name;

}

//4.输出查询结果

public static void printFruit(Fruit fruit) {

System.out.print("编号："+fruit.getId()+"\t\t");

System.out.print("名称："+fruit.getName()+"\t\t");

System.out.print("种类："+fruit.getType()+"\t");

System.out.print("单价："+fruit.getPrice()+"元/斤\t\t");

System.out.print("数量："+fruit.getQuantity()+"斤\n");

}

}

### （六）Text

package fruitManager;

public class Text {

public static void main(String[] args) {

// TODO 自动生成的方法存根

FruitDao\_imp fruit\_imp = new FruitDao\_imp();

while (true) {

int item = View.MenuView(); // 调用菜单视图

switch (item) {

case 0:

System.out.println("Exiting the application..."); // Debugging line

System.exit(0); // 退出程序

break; // 这里的 break 实际上不会执行，因为 System.exit(0) 已经退出了程序

case 1:

Fruit fruit = View.insertMenuView();

boolean insert = fruit\_imp.insert(fruit);

System.out.println(insert ? "添加成功\n" : "添加失败\n");

break; // 结束 case 1 的执行，返回循环开始，将再次显示菜单

case 2:

Fruit deleteMenuView = View.deleteMenuView();

boolean delete = fruit\_imp.delete(deleteMenuView);

System.out.println(delete ? "删除成功\n" : "删除失败\n");

break; // 结束 case 2 的执行

case 3:

Fruit updateMenuView = View.updateMenuView();

boolean update = fruit\_imp.update(updateMenuView);

System.out.println(update ? "修改成功\n" : "修改失败\n");

break; // 结束 case 3 的执行

case 4:

String selectMenuView = View.SelectMenuView();

Fruit select = fruit\_imp.select(selectMenuView);

View.printFruit(select);

break; // 结束 case 4 的执行

default:

System.out.println("Invalid option. Please try again.");

break; // 结束 default 的执行

}

// 注意：这里没有代码会执行，因为 break 语句会导致循环重新开始，再次调用 View.MenuView()

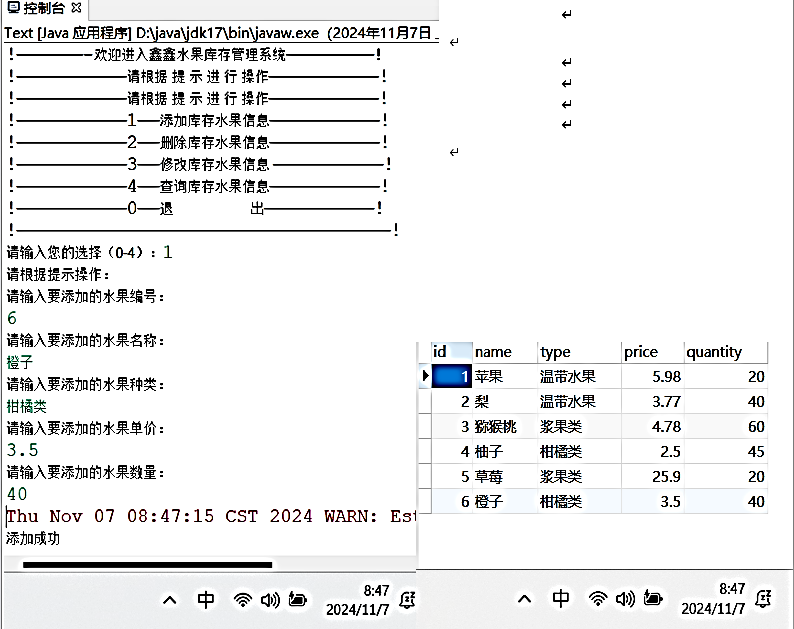
}

}

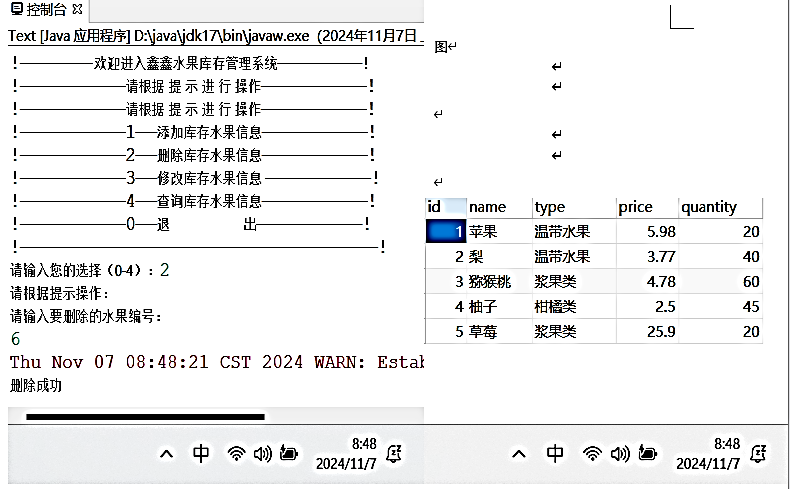
}

## 三、运行结果

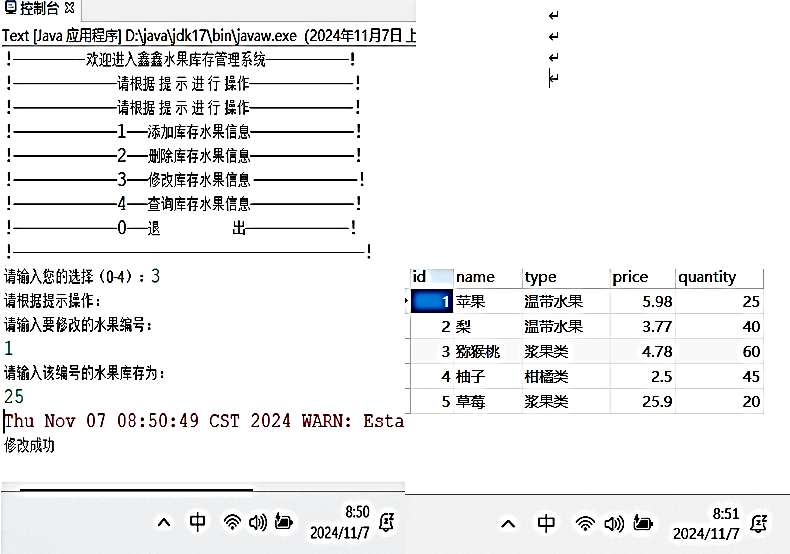
### （一）输入1（添加）



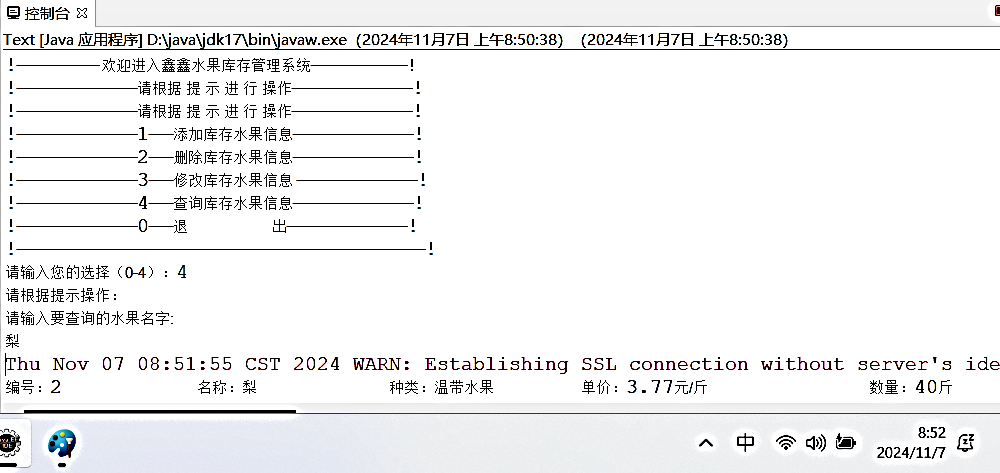
### （二）输入2（删除）



### （三）输入3（修改）



### （四）输入4（查询）



### （五）输入0（退出）

