

## Chuyên đề 6

### Bài tập tổng hợp

#### Mục tiêu

- ✓ Tổng hợp các kiến thức về Collection, File, JDBC trong Java Core 2.

#### Bài thực hành:

**Yêu cầu:** Viết ứng dụng quản lý sản phẩm thực hiện các công việc sau

**Câu 1:** Tạo cơ sở dữ liệu "**BkapShop**" có bảng "**tblShoes**" với các trường thông tin như sau

- **id:** kiểu dữ liệu số nguyên, tự tăng, là khóa chính, mã giày.
- **name:** kiểu chuỗi, giới hạn 32 ký tự là tên giày.
- **price:** kiểu số thực, là giá giày.
- **year:** năm sản xuất, kiểu dữ liệu số nguyên (vd: 2017)
- **description:** kiểu chuỗi giới hạn 64 ký tự là mô tả giày.

Tạo các Store Procedure có tên và chức năng như yêu cầu sau:

- **getAllShoes:** lấy về tất cả thông tin giày có trong bảng tblShoes, có sắp xếp theo tên a-z.
- **getShoesByName:** có tham số truyền vào là tên giày, hàm thủ tục này sẽ lấy về thông tin giày có chứa tên như tham số truyền vào (vd gợi ý: sử dụng LIKE '%Nike%').
- **addShoes:** nhận các tham số name, price, year, description để thực hiện insert dữ liệu vào trong bảng "tblShoes". Hàm thủ tục này trả về chuỗi "OK" nếu thành công.

**Câu 2:** Tạo lớp "**Shoes**" có

- Các trường dữ liệu tương ứng với các cột trong bảng "tblShoes" trên.
- Viết các constructor và hàm get/set.
- Viết phương thức nhập dữ liệu cho đối tượng
- Viết phương thức hiển thị dữ liệu đối tượng

**Câu 3:** Tạo lớp "**ShoesManager**" kết nối cơ sở dữ liệu và thực hiện các chức năng như sau

- Lấy tất cả dữ liệu giày trong cơ sở dữ liệu (sử dụng procedure **getAllShoes**)
- Tìm tất cả dữ liệu giày theo tên (sử dụng procedure **getShoesByName**)
- Thêm giày mới (sử dụng procedure **addShoes**)

**Câu 4:** Tạo class MainApplication có menu chức năng như sau

.: MENU .:

1. Thêm giày mới
2. Hiển thị tất cả giày
3. Tìm giày theo tên
4. Ghi dữ liệu ra file
5. Import dữ liệu
6. Thoát

Trong đó:

- Chức năng 1: thêm giày vào cơ sở dữ liệu
- Chức năng 2: tải toàn bộ thông tin từ csdl và hiển thị
- Chức năng 3: yêu cầu nhập tên giày và tìm kiếm trong csdl rồi hiển thị
- Chức năng 4: tải dữ liệu thông tin bảng "tblShoes" và ghi ra file dạng object
- Chức năng 5: đọc dữ liệu text từ file "data.txt" có thông tin như sau rồi thêm vào cơ sở dữ liệu.

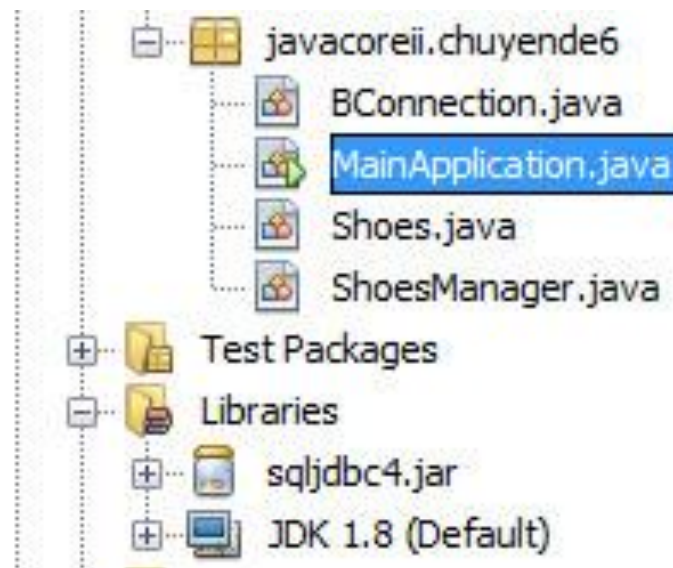
*Giày nam sneaker,120,2015,Giày vải bò*

*Giày da nữ,250,2016,Loại cao cổ lót bông*

```
Giày nam seaker,120,2015,Giày vải bò
Giày da nữ,250,2016,Loại cao cổ lót bông
```

File *data.txt*

**Câu 5:** Trong class MainApplication viết các hàm chức năng tương ứng với menu



```

Output - JavaCoreII.ChuyenDe (run)
Mời chọn từ 1 -> 6
2
Kết nối thành công
== Danh sách giày ==
Shoes{id=9, name=Bot co long, price=125.0, year=2018, description=}
Shoes{id=3, name=Bot da bò cao cấp, price=530.0, year=2016, description=Quý bà U50}
Shoes{id=11, name=Giày da nữ, price=250.0, year=2016, description=Loại cao cổ lót bông}
Shoes{id=10, name=Giày nam sneaker, price=120.0, year=2015, description=Giày vải bò}
Shoes{id=1, name=Giày vải, price=100.0, year=2012, description=Giày da năng}
Shoes{id=2, name=Guốc cao gót, price=120.0, year=1999, description=Dành cho teen}
Shoes{id=4, name=Guốc mộc, price=220.0, year=2010, description=Đồ cổ}
Shoes{id=5, name=Guốc mộc, price=220.0, year=2010, description=Đồ cổ}
Shoes{id=8, name=Guốc mộc, price=220.0, year=2010, description=Đồ cổ}
.: MENU :.
1. Thêm giày mới
2. Hiển thị tất cả
3. Tìm giày
4. Ghi dữ liệu ra file
5. Import dữ liệu
6. Thoát
Mời chọn từ 1 -> 6
    
```

## Code tham khảo:

### Shoes.java

`package` javacoreii.chuyende6;

`import` java.io.Serializable;

`import` java.util.Scanner;

/\*\*

\*

\* @author minhvuvc

\*/

```
public class Shoes implements Serializable {

    private int id;           // Khóa chính
    private String name;      // Tên sản phẩm
    private float price;      // Giá sản phẩm
    private int year;         // Năm sản xuất
    private String description; // Mô tả

    public Shoes() {
    }

    public Shoes(int id, String name, float price, int year, String description) {
        this.id = id;
        this.name = name;
        this.price = price;
        this.year = year;
        this.description = description;
    }

    public Shoes(String name, float price, int year, String description) {
        this.name = name;
        this.price = price;
        this.year = year;
        this.description = description;
    }

    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 float getPrice() {
        return price;
    }
}
```

```

    }

    public void setPrice(float price) {
        this.price = price;
    }

    public int getYear() {
        return year;
    }

    public void setYear(int year) {
        this.year = year;
    }

    public String getDescription() {
        return description;
    }

    public void setDescription(String description) {
        this.description = description;
    }

    /**
     * Hàm nhập dữ liệu cho đối tượng
     */
    public void input() {
        Scanner s = new Scanner(System.in);
        System.out.println("Tên sản phẩm: ");
        this.name = s.nextLine();
        System.out.println("Giá thành: ");
        this.price = s.nextFloat();
        System.out.println("Năm sản xuất: ");
        this.year = s.nextInt();
        System.out.println("Mô tả: ");
        this.description = s.nextLine();
        System.out.println("\t>>> Cảm ơn! <<<");
    }

    /**
     * Hàm xuất dữ liệu của đối tượng
     */
    public void output() {
        System.out.println("Shoes{ "
            + "id=" + id

```

```

        + ", name=" + name
        + ", price=" + price
        + ", year=" + year
        + ", description=" + description + '});
    }
}

```

## BConnection.java

```

package javacoreii.chuyende6;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;

/**
 *
 * @author minhvufc
 */
public class BConnection {

    private static final String HOST = "jdbc:sqlserver://localhost\\MINHVUFC-PC\\SQLEXPRESS2014:1433;databaseName=BkapShop";
    private static final String USERNAME = "sa";
    private static final String PASSWORD = "1234$";

    public static Connection getConnection() {
        try {
            Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
            return DriverManager.getConnection(HOST, USERNAME, PASSWORD);
        } catch (ClassNotFoundException ex) {
            Logger.getLogger(BConnection.class.getName()).log(Level.SEVERE, null, ex);
        } catch (SQLException ex) {
            Logger.getLogger(BConnection.class.getName()).log(Level.SEVERE, null, ex);
        }
        return null;
    }
}

```

## ShoesManager.java

```
package javacoreii.chuyende6;

import java.sql.CallableStatement;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Types;
import java.util.ArrayList;
import java.util.logging.Level;
import java.util.logging.Logger;

/**
 *
 * @author minhvufc
 */
public class ShoesManager {

    private Connection conn;
    ArrayList<Shoes> lstShoes = new ArrayList<>();

    public ShoesManager() {
        conn = BConnection.getConnection();
        if (conn != null) {
            System.out.println("Kết nối thành công");
        } else {
            System.err.println(">>> Kiểm tra kết nối !!!");
        }
    }

    public void getAllShoes() {
        // Reset mảng dữ liệu
        lstShoes.removeAll(lstShoes);
        try {
            String sql = "{CALL getAllShoes}";
            CallableStatement cs = conn.prepareCall(sql);
            ResultSet rs = cs.executeQuery();
            while (rs.next()) {
                Shoes s = new Shoes(rs.getInt("id"),
                    rs.getString("name"), rs.getFloat("price"),
                    rs.getInt("year"), rs.getString("description"));
                lstShoes.add(s);
                s.output(); // In dữ liệu
            }
            if (lstShoes.size() == 0) {

```

```

        System.out.println("KHÔNG TÌM THẤY DỮ LIỆU");
    }
} catch (SQLException ex) {
    Logger.getLogger(ShoesManager.class.getName()).log(Level.SEVERE,
null, ex);
}
}

public void getShoesByName(String name) {
    // Reset mảng dữ liệu
    lstShoes.removeAll(lstShoes);
    try {
        String sql = "{CALL getShoesByName(?)}";
        CallableStatement cs = conn.prepareCall(sql);
        cs.setString(1, name);
        ResultSet rs = cs.executeQuery();
        while (rs.next()) {
            Shoes s = new Shoes(rs.getInt("id"),
                rs.getString("name"), rs.getFloat("price"),
                rs.getInt("year"), rs.getString("description"));
            lstShoes.add(s);
            s.output(); // In dữ liệu
        }
        if (lstShoes.size() == 0) {
            System.out.println("KHÔNG TÌM THẤY DỮ LIỆU");
        }
    } catch (SQLException ex) {
        Logger.getLogger(ShoesManager.class.getName()).log(Level.SEVERE,
null, ex);
    }
}

public void create() {
    try {
        Shoes s = new Shoes();
        s.input();
        String sql = "{CALL addShoes(?,?,?,?,?)}";
        CallableStatement cs = conn.prepareCall(sql);
        cs.setString(1, s.getName());
        cs.setFloat(2, s.getPrice());
        cs.setInt(3, s.getYear());
        cs.setString(4, s.getDescription());
        cs.registerOutParameter(5, Types.NVARCHAR);
    }
}

```



```

        cs.executeUpdate();
        // Lấy kết quả trả về
        String result = cs.getString(5);
        System.out.println("Thêm dữ liệu: " + result);
    } catch (SQLException ex) {
        Logger.getLogger(ShoesManager.class.getName()).log(Level.SEVERE,
null, ex);
    }
}

public void importShoes(Shoes s) {
    try {
        String sql = "{CALL addShoes(?,?,?,?,?)}";
        CallableStatement cs = conn.prepareCall(sql);
        cs.setString(1, s.getName());
        cs.setFloat(2, s.getPrice());
        cs.setInt(3, s.getYear());
        cs.setString(4, s.getDescription());
        cs.registerOutParameter(5, Types.NVARCHAR);

        cs.executeUpdate();
        // Lấy kết quả trả về
        String result = cs.getString(5);
        System.out.println("Thêm dữ liệu: " + result);
    } catch (SQLException ex) {
        Logger.getLogger(ShoesManager.class.getName()).log(Level.SEVERE,
null, ex);
    }
}
}

```

## MainApplication.java

```

package javacoreii.chuyende6;

import java.io.BufferedReader;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.ObjectOutputStream;
import java.util.Scanner;
import java.util.logging.Level;
import java.util.logging.Logger;

```

```
/**
 *
 * @author minhvuvc
 */
public class MainApplication {

    private int showMenu() {
        System.out.println("\t.: MENU .:");
        System.out.println("1. Thêm giày mới");
        System.out.println("2. Hiển thị tất cả");
        System.out.println("3. Tìm giày");
        System.out.println("4. Ghi dữ liệu ra file");
        System.out.println("5. Import dữ liệu");
        System.out.println("6. Thoát");
        System.out.println("Mời chọn từ 1 -> 6");
        Scanner s = new Scanner(System.in);
        return s.nextInt();
    }

    private void add() {
        ShoesManager sm = new ShoesManager();
        // Thêm
        sm.create();
    }

    private void read() {
        ShoesManager sm = new ShoesManager();
        System.out.println("== Danh sách giày ==");
        sm.getAllShoes();
    }

    private void readBy() {
        Scanner s = new Scanner(System.in);
        System.out.println("Nhập tên giày cần tìm: ");
        String shoesName = s.nextLine();

        System.out.println("== Danh sách giày <" + shoesName + "> ==");
        ShoesManager sm = new ShoesManager();
        sm.getShoesByName(shoesName);
    }

    private void writeData() {
        FileOutputStream fos = null;
    }
}
```

```

    try {
        String filePath = "shoes.db";
        fos = new FileOutputStream(filePath);
        ObjectOutputStream oos = new ObjectOutputStream(fos);

        ShoesManager sm = new ShoesManager();
        sm.getAllShoes(); // Lấy dữ liệu từ database
        oos.writeObject(sm.lstShoes); // Ghi xuống file
        oos.flush();

        oos.close();
        fos.close();
        System.out.println("Ghi dữ liệu thành công");

    } catch (FileNotFoundException ex) {
        Logger.getLogger(MainApplication.class.getName()).log(Level.SEVERE,
null, ex);
    } catch (IOException ex) {
        Logger.getLogger(MainApplication.class.getName()).log(Level.SEVERE,
null, ex);
    } finally {
        try {
            fos.close();
        } catch (IOException ex) {

        }
    }

    Logger.getLogger(MainApplication.class.getName()).log(Level.SEVERE, null, ex);
}
}

private void importData() {
    FileInputStream fis = null;
    try {
        String filePath = "data.txt";
        fis = new FileInputStream(filePath);
        BufferedReader br = new BufferedReader(new InputStreamReader(fis));
        String line;
        while ((line = br.readLine()) != null) {
            String row[] = line.split(",");
            Shoes s = new Shoes(row[0], Float.parseFloat(row[1]),
Integer.parseInt(row[2]), row[3]);
            ShoesManager sm = new ShoesManager();
            sm.importShoes(s);
        }
    }
}

```

```
        System.out.println("Import dữ liệu thành công");
    } catch (FileNotFoundException ex) {
        Logger.getLogger(MainApplication.class.getName()).log(Level.SEVERE,
null, ex);
    } catch (IOException ex) {
        Logger.getLogger(MainApplication.class.getName()).log(Level.SEVERE,
null, ex);
    } finally {
        try {
            fis.close();
        } catch (IOException ex) {

Logger.getLogger(MainApplication.class.getName()).log(Level.SEVERE, null, ex);
        }
    }

}

/**
 * @param args the command line arguments
 */
public static void main(String[] args) {
    MainApplication main = new MainApplication();
    int luaChon = 0;
    do {
        luaChon = main.showMenu();
        switch (luaChon) {
            case 1:
                main.add();
                break;
            case 2:
                main.read();
                break;
            case 3:
                main.readBy();
                break;
            case 4:
                main.writeData();
                break;
            case 5:
                main.importData();
                break;
            case 6:
                System.out.println("Chào tạm biệt");
```

```
        break;
    }
} while (luaChon > 0 && luaChon < 6);
System.out.println("Chương trình kết thúc");
}
}
```