## 使用dom4j解析xml文件

### 步骤分析：

1. 导入 jar 包

dom4j-1.6.1.jar

1. 创建一个核心对象

SAXReader reader = new SAXReader();

1. 将 xml 文档加载到内存中形成一棵树

Document doc = reader.read("F:\\projects\\java\\day08\\xml\\web.xml");

1. 获取根节点

Element root=doc.getRootElement();

1. 通过根节点获取其他节点（文本节点、属性节点、元素节点）

List<Element> list = root.elements();

//遍历集合

for (Element ele : list) {

//获取servlet-name的标签体

String text = ele.elementText("servlet-name");

System.out.println(text);

//获取url-pattern标签体

System.out.println(ele.elementText("url-pattern"));

}

//获取root的version属性值

String value = root.attributeValue("version");

System.out.println(value);

### 实现：

1. web.xml 路径：F:\\projects\\java\\day08\\xml\\web.xml

<?xml version="1.0" encoding="UTF-8"?>

<web-app version="2.5">

<servlet>

<servlet-name>HelloMyServlet</servlet-name>

<servlet-class>com.itheima.HelloMyServlet</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>HelloMyServlet</servlet-name>

<url-pattern>/hello</url-pattern>

</servlet-mapping>

</web-app>

2. Dom4jDemo.java

import java.util.List;

import org.dom4j.Document;

import org.dom4j.DocumentException;

import org.dom4j.Element;

import org.dom4j.io.SAXReader;

public class Dom4jDemo {

public static void main(String[] args) throws Exception {

//创建核心对象

SAXReader reader = new SAXReader();

//获取dom树

Document doc = reader.read("F:\\projects\\java\\day08\\xml\\web.xml");

//获取根节点

Element root=doc.getRootElement();

//获取其他节点

List<Element> list = root.elements();

//遍历集合

for (Element ele : list) {

//获取servlet-name的标签体

String text = ele.elementText("servlet-name");

System.out.println(text);

//获取url-pattern标签体

System.out.println(ele.elementText("url-pattern"));

}

//获取root的version属性值

String value = root.attributeValue("version");

System.out.println(value);

}

}

## 登录案例

### 需求:

在页面上输入用户名和密码,提交到服务器上,服务器拿着用户名和密码去数据库中查找有无此用户

若有用户,则提示登录成功

若无此用户,则提示用户名密码不匹配

### 技术分析:

表单

servlet

请求(request)

响应(response)

### 步骤分析：

1.先有数据库和表

create database day09;

use day09;

create table user(

id int primary key auto\_increment,

username varchar(20),

password varchar(20),

email varchar(20),

name varchar(20),

sex varchar(10),

birthday date,

hobby varchar(50)

);

insert into user values (null,'tom','123','tom@126.com','tom','1','1988-01-01',null);

2.创建工程

3.复制页面

修改login.html

给用户名和密码添加name属性

修改表单的action属性

action="http://localhost/day0901/login"

添加method属性

method="post"

4.导入jar包:

驱动 dbutils c3p0

5.导入工具类和配置文件

datasourceUtils

c3p0-config.xml(注意修改数据名称)

6.创建servlet(LoginServlet: 路径 /login)

接受用户名和密码

调用service层(UserService)完成登录操作

提示信息

7.UserService

login(username,password)

调用dao

8.dao

通过用户名和密码查询数据库

### 实现：

1. 创建数据库和表

create database day09;

use day09;

create table user(

id int primary key auto\_increment,

username varchar(20),

password varchar(20),

email varchar(20),

name varchar(20),

sex varchar(10),

birthday date,

hobby varchar(50)

);

insert into user values (null,'tom','123','tom@126.com','tom','1','1988-01-01',null);

1. 创建Login项目
2. 导包
   1. 驱动
   2. dbutils
   3. c3p0
3. 导入工具类和配置文件
   1. datasourceUtils
   2. c3p0-config.xml(注意修改数据名称)
4. 编写login.html
5. 创建和配置LoginServlet