武汉纺织大学

Web应用开发课程设计

# 新闻展示

**学 院： 数学与计算机学院**

**班 级： 物联网11802班**

**姓 名： 张新龙**

**学 号： 1804241018**

**指导老师： 聂刚**

**成 绩：**

**完成日期： 2020年12月15日**

1. **需求分析**



按上图设计实现新闻显示页面，页面所需图片，文字均来自数据库。

1. **系统设计**

**数据库**

**A）新闻数据库**

SET FOREIGN\_KEY\_CHECKS=0;

-- ----------------------------

-- Table structure for news\_info

-- ----------------------------

DROP TABLE IF EXISTS `news\_info`;

CREATE TABLE `news\_info` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`title` varchar(255) DEFAULT NULL COMMENT '新闻标题',

`date` datetime DEFAULT NULL COMMENT '新闻时间',

`content` varchar(255) DEFAULT NULL COMMENT '新闻内容',

`type` int(11) DEFAULT NULL COMMENT '类型:1:人事信息2:教务信息3:学工信息',

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=17 DEFAULT CHARSET=utf8;

-- ----------------------------

-- Records of news\_info

-- ----------------------------

INSERT INTO `news\_info` VALUES ('1', '2020-2021-1学期南湖校区“戏剧与影视学类全', '2020-12-18 10:04:06', '1', '3');

INSERT INTO `news\_info` VALUES ('2', '关于开展2020-2021（1）期末教学评价的通知', '2020-12-22 10:04:18', '2', '3');

INSERT INTO `news\_info` VALUES ('3', '武汉纺织大学2018、2019级辅修双学位学生缴费', '2020-12-16 13:05:35', null, '3');

INSERT INTO `news\_info` VALUES ('4', '关于提交2020年度工程教育认证持续改进情况报', '2020-12-18 13:05:40', null, '3');

INSERT INTO `news\_info` VALUES ('5', '关于举办武汉纺织大学2020年本科生学术论坛（', '2020-12-16 13:05:43', null, '3');

INSERT INTO `news\_info` VALUES ('6', '关于开展武汉纺织大学2020年本科专业建设负责', '2020-12-10 13:05:47', null, '3');

INSERT INTO `news\_info` VALUES ('7', '武汉纺织大学2020年教师信息化培训服务项目公', '2020-12-08 13:05:52', null, '3');

INSERT INTO `news\_info` VALUES ('8', '武汉纺织大学2019年说课视频拍摄制作服务项目', '2020-12-12 13:05:56', null, '3');

INSERT INTO `news\_info` VALUES ('9', '学校举办首届本科生学术论坛（第三场）', '2020-12-09 13:06:08', null, '3');

INSERT INTO `news\_info` VALUES ('10', '本科生学术论坛（第二场）在阳光校区举办', '2020-12-14 13:06:14', null, '3');

INSERT INTO `news\_info` VALUES ('11', '教务处组织“突出教育教学实绩，践行教书育人', '2020-12-07 13:06:20', null, '3');

INSERT INTO `news\_info` VALUES ('12', '学校举办首届本科生学术论坛（第一场）', '2020-12-04 13:06:25', null, '3');

INSERT INTO `news\_info` VALUES ('13', '教务处召开教学秘书业务培训及工作交流会', '2020-12-15 13:06:28', null, '3');

INSERT INTO `news\_info` VALUES ('14', '郑用琏教授应邀来我校做教师岗位能力提升培训', '2020-12-14 13:06:33', null, '3');

INSERT INTO `news\_info` VALUES ('15', '学校到京博控股集团考察对接学生实习实训工作', '2020-12-17 13:06:38', null, '3');

INSERT INTO `news\_info` VALUES ('16', '学校举行新任本科教学督导组暨课程思政专家', '2020-12-14 13:06:44', null, '3');

**B）新闻相关数据库**

SET FOREIGN\_KEY\_CHECKS=0;

-- ----------------------------

-- Table structure for relation

-- ----------------------------

DROP TABLE IF EXISTS `relation`;

CREATE TABLE `relation` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`name` varchar(255) DEFAULT NULL COMMENT '名称',

`icon` varchar(255) CHARACTER SET utf8 COLLATE utf8\_general\_ci DEFAULT NULL COMMENT '图标信息',

`type` int(11) DEFAULT NULL COMMENT '类型1是专题网站2是资源中心',

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=12 DEFAULT CHARSET=utf8;

-- ----------------------------

-- Records of relation

-- ----------------------------

INSERT INTO `relation` VALUES ('1', '校庆专题', 'glyphicon glyphicon-home', '1');

INSERT INTO `relation` VALUES ('2', 'VPN', 'glyphicon glyphicon-collapse-down', '1');

INSERT INTO `relation` VALUES ('3', '图书资源', 'glyphicon glyphicon-book', '1');

INSERT INTO `relation` VALUES ('4', '招标网', 'glyphicon glyphicon-modal-window', '1');

INSERT INTO `relation` VALUES ('5', '信息公开', 'glyphicon glyphicon-comment', '1');

INSERT INTO `relation` VALUES ('6', '人才招聘', 'glyphicon glyphicon-modal-window', '1');

INSERT INTO `relation` VALUES ('7', '网络课堂', 'glyphicon glyphicon-play-circle', '1');

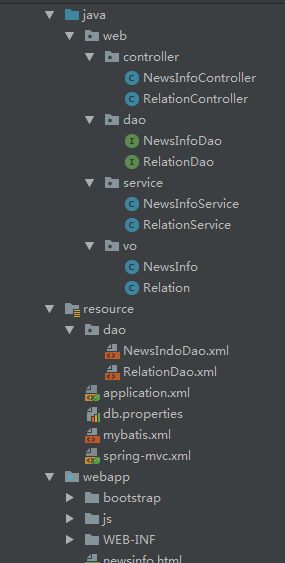
INSERT INTO `relation` VALUES ('8', '邮件系统', 'glyphicon glyphicon-envelope', '1');

INSERT INTO `relation` VALUES ('9', '电子图书馆', null, '2');

INSERT INTO `relation` VALUES ('10', '网络资源', null, '2');

INSERT INTO `relation` VALUES ('11', '网络教学平台', null, '2');

**项目结构**

****

**配置文件**

**db.properties文件**

该配置文件主要是以文件形式保存数据库的驱动类名称，连接数据库的URL地址，访问数据库的用户名及对应的密码，程序运行时会读取该文件相关信息，避免硬编码，当相关信息发生变化时，只需修改配置文件而不用修改源代码，增加程序的可扩展型。

jdbc.driver=com.mysql.jdbc.Driver

jdbc.url=jdbc:mysql://127.0.0.1/project\_zxl?characterEncoding=utf-8

jdbc.username=root

jdbc.password=123456

**mybatis.xml文件**

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

<!DOCTYPE configuration

        PUBLIC "-//mybatis.org//DTD Config 3.0//EN"

        "http://mybatis.org/dtd/mybatis-3-config.dtd">

<configuration>

*<!--打印sql-->*

    <settings>

        <setting name="logImpl" value="STDOUT\_LOGGING"/>

    </settings>

    <typeAliases>

*<!--包扫描-->*

        <package name="web.vo"/>

    </typeAliases>

</configuration>

**实体类NEWS**

package web.vo;

import com.fasterxml.jackson.annotation.JsonFormat;

import java.util.Date;

*// 新闻信息实体*

public class NewsInfo {

    private Integer id;

*// 新闻标题*

    private String title;

*// 新闻发布时间*

    @JsonFormat(pattern = "yyyy-MM-dd", timezone="GMT+8")

    private Date date;

*// 新闻内容*

    private String content;

*// 类型:1:人事信息2:教务信息3:学工信息*

    private Integer type;

    public Integer getId() {

        return id;

    }

    public void setId(Integer *id*) {

        this.id = id;

    }

    public String getTitle() {

        return title;

    }

    public void setTitle(String *title*) {

        this.title = title;

    }

    public Date getDate() {

        return date;

    }

    public void setDate(Date *date*) {

        this.date = date;

    }

    public String getContent() {

        return content;

    }

    public void setContent(String *content*) {

        this.content = content;

    }

    public Integer getType() {

        return type;

    }

    public void setType(Integer *type*) {

        this.type = type;

    }

}

package web.vo;

*// 新闻关联实体*

public class Relation {

    private Integer id;

*// 名称*

    private String name;

*// 图标信息*

    private String icon;

*// 类型1是专题网站2是资源中心*

    private Integer type;

    public Integer getId() {

        return id;

    }

    public void setId(Integer *id*) {

        this.id = id;

    }

    public String getName() {

        return name;

    }

    public void setName(String *name*) {

        this.name = name;

    }

    public Integer getType() {

        return type;

    }

    public void setType(Integer *type*) {

        this.type = type;

    }

    public String getIcon() {

        return icon;

    }

    public void setIcon(String *icon*) {

        this.icon = icon;

    }

}

**接口类News**

package web.dao;

import web.vo.NewsInfo;

import java.util.List;

*// 新闻信息dao层*

public interface NewsInfoDao {

*// 根据类型查询新闻信息*

    List<NewsInfo> selectNewsInfoByType(Integer *type*);

}

package web.controller;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Controller;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.ResponseBody;

import web.service.NewsInfoService;

import web.vo.NewsInfo;

import java.util.List;

*// 新闻信息Controller层*

@Controller

public class NewsInfoController {

    @Autowired

    private NewsInfoService newsInfoService;

*// 根据类型查询新闻信息*

    @RequestMapping("/selectNewsInfoByType")

    @ResponseBody

    public List<NewsInfo> selectNewsInfoByType(Integer *type*){

*// 调用service层处理请求*

        List<NewsInfo> newsInfos = newsInfoService.selectNewsInfoByType(type);

        return newsInfos;

    }

}

**接口类News相关**

package web.dao;

import web.vo.Relation;

import java.util.List;

*// 关联表的dao层*

public interface RelationDao {

*// 根据类型查询数据*

    List<Relation> selectRelationByType(Integer *type*);

}

package web.controller;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Controller;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.ResponseBody;

import web.service.RelationService;

import web.vo.Relation;

import java.util.List;

*// 关联表controller*

@Controller

public class RelationController {

    @Autowired

    private RelationService relationService;

*// 根据类型查出关联信息*

    @RequestMapping("/selectRelationByType")

    @ResponseBody

    public List<Relation> selectRelationByType(Integer *type*){

*// 调用service层进行处理*

        List<Relation> relations = relationService.selectRelationByType(type);

*// 返回结果*

        return relations;

    }

}

**服务层NewsService**

package web.service;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

import web.dao.NewsInfoDao;

import web.vo.NewsInfo;

import java.util.List;

*// 新闻信息服务层*

@Service

public class NewsInfoService {

    @Autowired

    private NewsInfoDao newsInfoDao;

*// 根据新闻类型查询*

    public List<NewsInfo> selectNewsInfoByType(Integer *type*){

        List<NewsInfo> newsInfos = newsInfoDao.selectNewsInfoByType(type);

        return newsInfos;

    }

}

package web.service;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

import web.dao.RelationDao;

import web.vo.Relation;

import java.util.List;

*// 关联表service*

@Service

public class RelationService {

    @Autowired

    private RelationDao relationDao;

*// 根据类型查出关联表的信息*

    public List<Relation> selectRelationByType(Integer *type*) {

*// 调用dao层查询数据*

        List<Relation> relations = relationDao.selectRelationByType(type);

        return relations;

    }

}

**前端页面**

<!DOCTYPE html>

<html lang="en" xmlns:v-on="http://www.w3.org/1999/xhtml">

<head>

<meta charset="UTF-8">

<title>Title</title>

<link rel="stylesheet" href="bootstrap/css/bootstrap.min.css">

<script type="text/javascript" src="./js/jquery-3.4.1.min.js"></script>

<script type="text/javascript" src="./bootstrap/js/bootstrap.min.js"></script>

<script type="text/javascript" src="./js/vue.js"></script>

</head>

<style>

#app{

width: 1200px;

height: 400px;

margin: 0 auto;

}

#leftFloat{

width: 750px;

height: 400px;

float: left;

}

#rightFloat{

width: 450px;

height: 400px;

float: right;

padding-left: 100px;

}

.mytext{

overflow: hidden;

white-space: nowrap;

text-overflow: ellipsis;

}

</style>

<body>

<div id="app">

<div id="leftFloat">

<ul class="nav nav-tabs">

<li role="presentation" ><a href="#">人事信息</a></li>

<li role="presentation"><a href="#">教务系统</a></li>

<li role="presentation" class="active"><a href="#">学工信息</a></li>

</ul>

<div class="row" style="margin-top: 15px" v-for="(newsInfo, index) in newsInfos" v-if="index % 2 == 0">

<div class="col-md-4 mytext">{{newsInfo.title}}</div>

<div class="col-md-2">{{newsInfo.date}}</div>

<div class="col-md-4 mytext">{{newsInfos[index + 1].title}}</div>

<div class="col-md-2">{{newsInfos[index + 1].date}}</div>

</div>

</div>

<div id="rightFloat">

<h3>专题网站</h3>

<hr>

<div class="row" style="margin-top: 20px" v-for="(relation, index) in relation1" v-if="index % 2 == 0">

<div class="col-md-1"><span :class="relation.icon" aria-hidden="true"></span></div>

<div class="col-md-5">{{relation.name}}</div>

<div class="col-md-1"><span :class="relation1[index + 1].icon" aria-hidden="true"></span></div>

<div class="col-md-5">{{relation1[index + 1].name}}</div>

</div>

<h3>资源中心</h3>

<hr>

<div class="row" style="margin-top: 20px" v-for="(relation, index) in relation2" v-if="index % 2 == 0">

<div class="col-md-6" >{{relation.name}}</div>

<div class="col-md-6" v-if="index + 1 < relation2.length">{{relation2[index + 1].name}}</div>

</div>

</div>

</div>

</body>

<script type="text/javascript">

var vue = new Vue({

el: '#app',

data: {

newsInfos:[],

relation1:[],

relation2:[]

},

methods:{

// 页面加载时查询数据进行初始化

selectNewsInfos: function(){

$.ajax({

url:"/selectNewsInfoByType",

type:"GET",

data:{type : 3},

success:function(result){

vue.newsInfos = result;

}

});

},

selectRelation1:function () {

$.ajax({

url:"/selectRelationByType",

type:"GET",

data:{type : 1},

success:function(result){

vue.relation1 = result;

}

});

},

selectRelation2:function () {

$.ajax({

url:"/selectRelationByType",

type:"GET",

data:{type: 2},

success:function(result){

vue.relation2 = result;

}

});

}

},

create:function () {

},

mounted:function () {

// 利用vue钩子函数 在页面初始化的时候发送请求获取数据

this.selectNewsInfos();

this.selectRelation1();

this.selectRelation2();

}

})

</script>

</html>

1. **系统测试**

****

1. **系统总结**

**Web这门课程让我们对编写网页产生了浓厚的兴趣，每次上课看着老师给我们展示不同的功能时，就觉得这是一门很神奇的学科。因为这是一门实践性很强的学科，所以课后自己也会对照着老师的PPT，试着完成不同的功能，每次做成了也非常有成就感。Web的前半部分，介绍HTML、CSS、JavaScript的部分相对来说比较简单，而后半部分关于web与数据库连接的内容有点困难，最开始做的时候一直出错。直到后来在老师和同学的帮助下才慢慢解决问题。 由于平时学习比较繁忙，时间比较少，对于网页设计软件的强大功能运用得还不够。在网页设计过程中，由于缺乏训练，所以在设计时也遇到了很多麻烦，有时插入一个特效脚本也是很麻烦。在网页制作过程中，想提高设计水平，想有个理想的网页，必须要勤奋练习，我将在以后的学习过程中不断努力。 结语 总体来说，这学期的Web学习有好的部分，也有不足的部分。但是至少自己已经入门，能够独立编写一些网页了。**