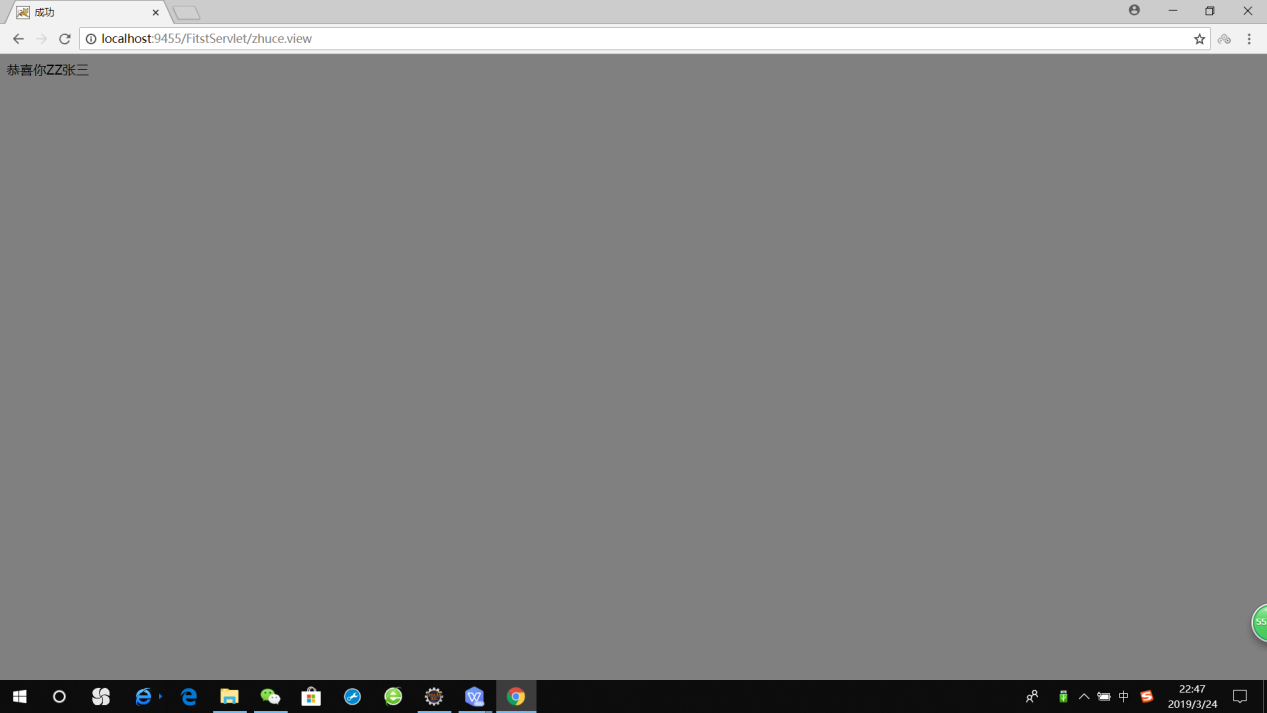
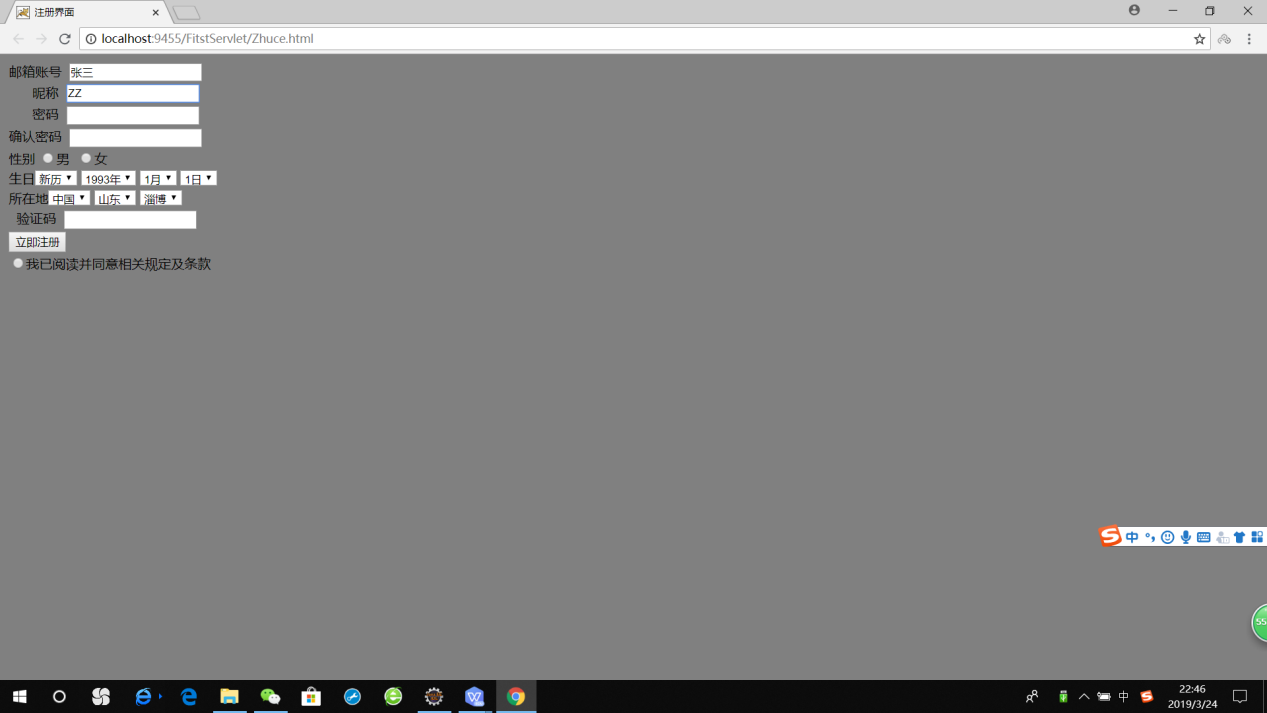
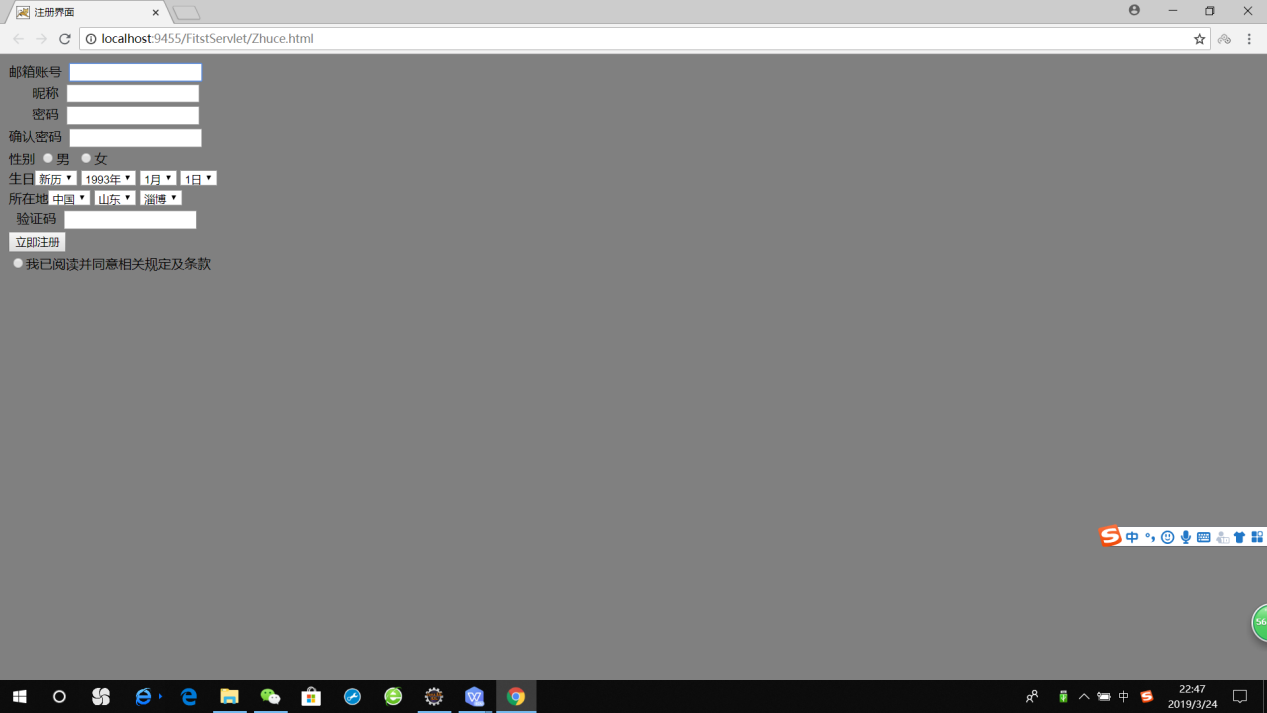
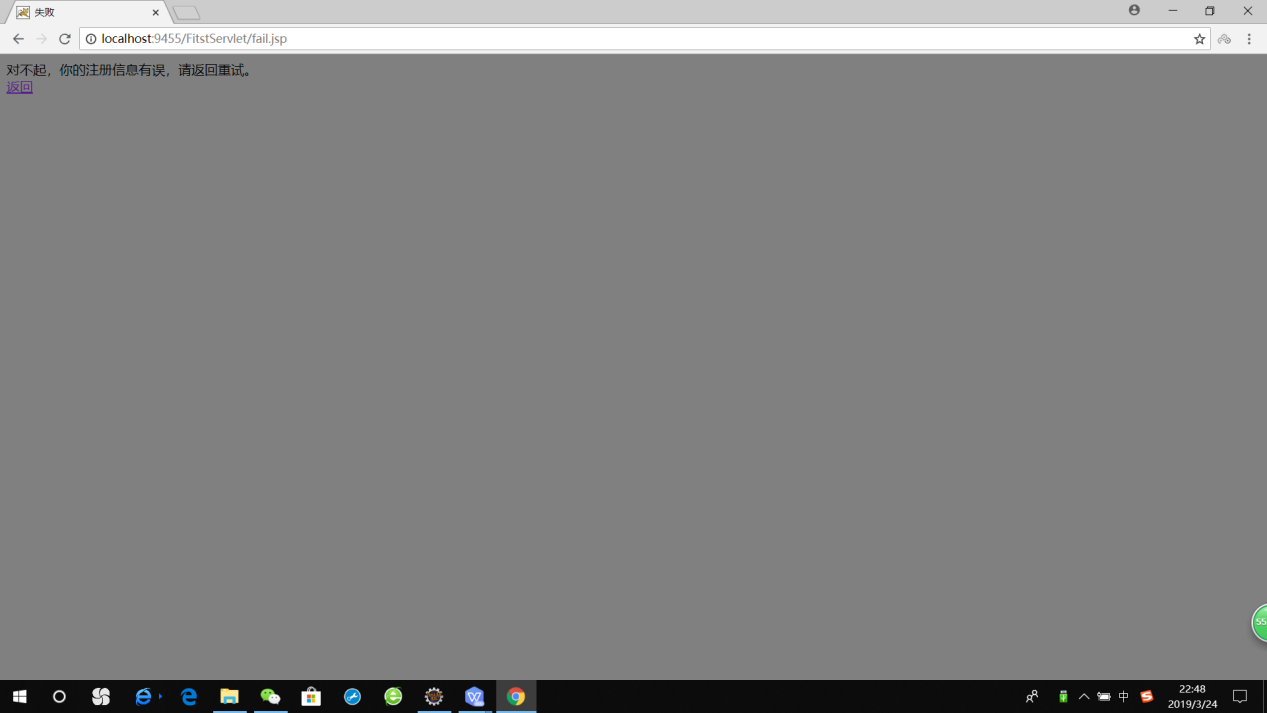
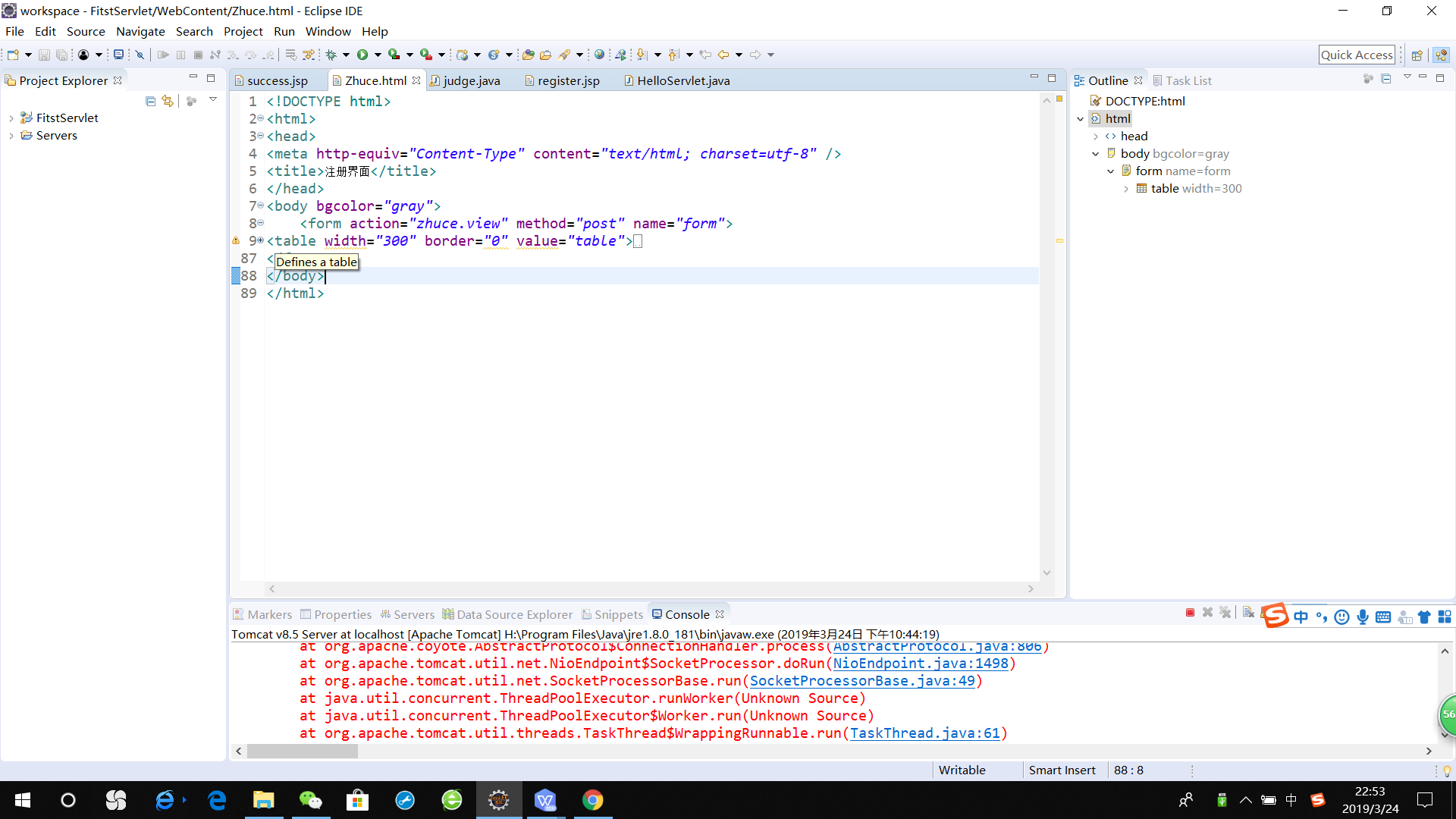
* 上一周做的注册页面也可，验证条件可以自行设定，注册之后根据用户输入情况进行相应的提示。









***Judge.java***

package Register;

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

/\*\*

\* Servlet implementation class judge

\*/

@WebServlet("/zhuce.view")

public class judge extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public judge() {

super();

// TODO Auto-generated constructor stub

}

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

doPost(request,response);

}

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

// TODO Auto-generated method stub

request.setCharacterEncoding("UTF-8");

response.setCharacterEncoding("UTF-8");

PrintWriter out =response.getWriter();

response.setContentType("text/html;charset=UTF-8");

String name=request.getParameter("name");

String zhanghao=request.getParameter("zhanghao");

request.setAttribute("name",name);

request.setAttribute("zhanghao",zhanghao);

if(name.length()!=0&&zhanghao.length()!=0) {

// response.sendRedirect("success.jsp");

request.getRequestDispatcher("success.jsp").forward(request,response);

}

else {

response.sendRedirect("fail.jsp");

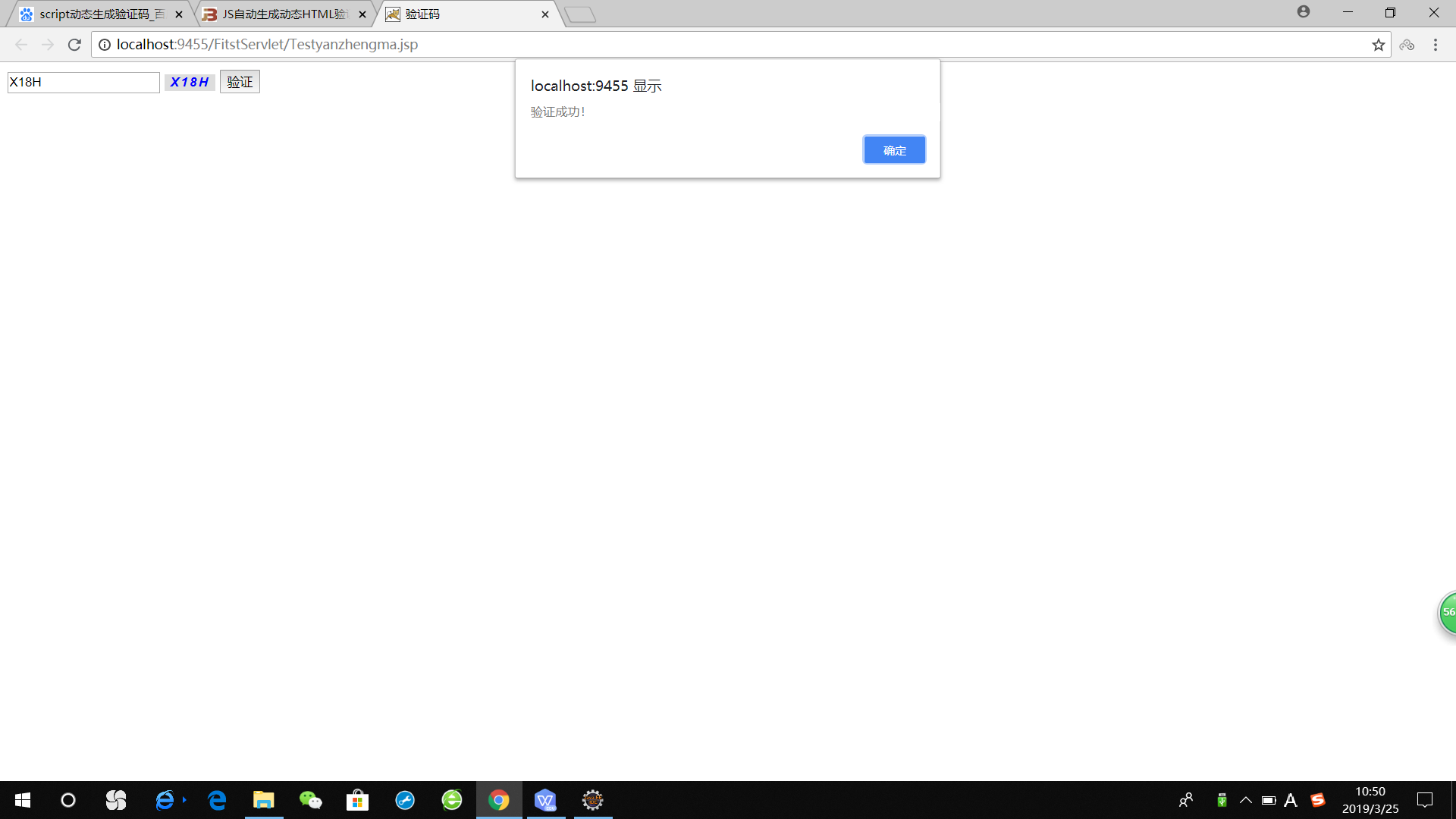
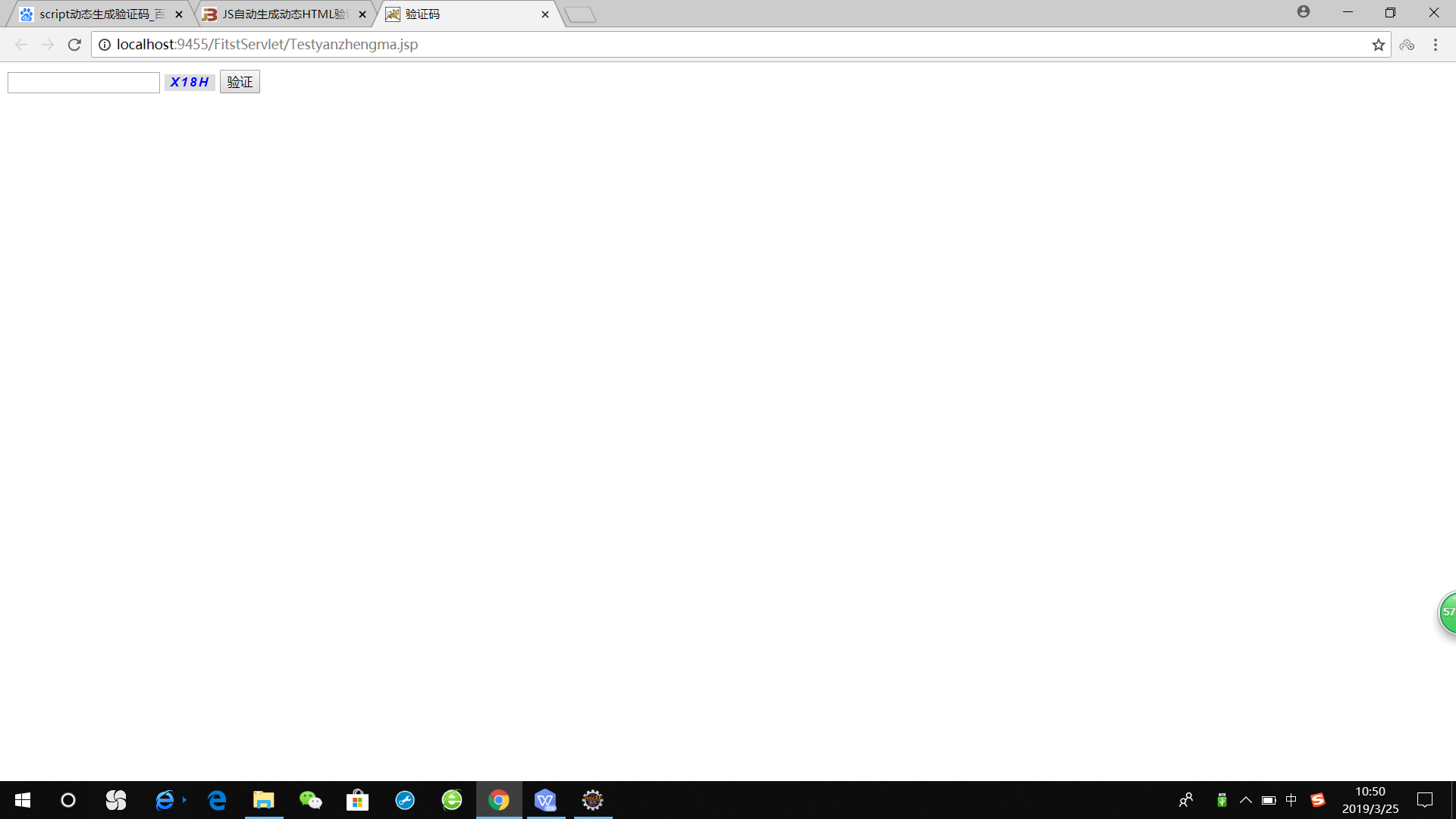
}

}

}

课后第三章第一题。动态生成验证码（选做，生成动态验证码的方法请自行百度）

利用Servlet动态生成验证码，注意只需实现动态产出密码图片即可，暂时不需要进行验证。如生成图片比较困难，则只需生成动态字符串并显示在页面上。



代码：

<%@ page contentType=*"text/html;charset=UTF-8"* language=*"java"* import=*"java.util.\*"* %>

<!DOCTYPE html>

<html>

<head>

<title>验证码</title>

<meta charset=*"utf-8"* />

<style type=*"text/css"*>

*#code* {

font-family: *Arial*;

font-style: *italic*;

font-weight: *bold*;

border: *0*;

letter-spacing: *2px*;

color: *blue*;

}

</style>

<script>

//产生验证码

window.onload = **function**() {

createCode()

}

**var** code; //在全局定义验证码

**function** createCode() {

code = "";

**var** codeLength = 4; //验证码的长度

**var** checkCode = document.getElementById("code");

**var** random = **new** Array(0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R',

'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z'); //随机数

**for**(**var** i = 0; i < codeLength; i++) { //循环操作

**var** index = Math.floor(Math.random() \* 36); //取得随机数的索引（0~35）

code += random[index]; //根据索引取得随机数加到code上

}

checkCode.value = code; //把code值赋给验证码

}

//校验验证码

**function** validate() {

**var** inputCode = document.getElementById("ctl00\_txtcode").value.toUpperCase(); //获取输入框内验证码并转化为大写

**if**(inputCode.length <= 0) { //若输入的验证码长度为0

alert("请输入验证码！"); //则弹出请输入验证码

}

**else** **if**(inputCode != code) { //若输入的验证码与产生的验证码不一致时

alert("验证码输入错误!"); //则弹出验证码输入错误

createCode(); //刷新验证码

document.getElementById("ctl00\_txtcode").value = "";//清空文本框

} **else** { //输入正确时

alert("验证成功！"); //弹出

}

}

</script>

</head>

<body>

<div>

<!--时间：2017-01-11 描述：输入框ct100\_textcode -->

<input type=*"text"* id=*"ctl00\_txtcode"* />

<!--时间：2017-01-11 描述：把验证码定义为按钮，点击刷新-->

<input type=*"button"* id=*"code"* onclick="createCode()" />

<!--时间：2017-01-11 描述：验证按钮 -->

<input type=*"button"* value=*"验证"* onclick="validate()" />

</div>

</body>

</html>