step2 3주차

- 지금 대시보드.html에 있는 내용이 디폴트 레이아웃.jsp에 들어가 있어야 함으로

먼저 전부를 복사하여 입력한다.

- 언어설정이 영어로 되어 잇는 것을 삭제

- 각각 클래스별로 구분하여 마이너스 표시누름

- 지금 레이앙웃이 있는 5가지를 제외한 나머지부분은 그대로 두면 된다.

----- 위 부분은 ppt 통해서 복습할 것 !!!

- 레프트 메뉴에 있는 버튼 클릭 - css깨짐

: 왜냐하면 그버튼을 누르면 <http://localhost:8080/s22119/dashboard.html> 여기를 가르키기 때문

그렇다면 jsp 를 보도록 하면 되기 때문에 default-left.jsp에서 dashboard.html로 된 것을 main.do로 바꿈 // 스프링에서 jsp에서 jsp를 직접 호출 못함

<a href="#" onclick="pageSubmitFn()">

- href에 #을 적으면 자기자신 호출 // onclick 속성은 내가 클릭하면 속성안에 있는 값이 실행 되는 것 (여기서 무엇을 클릭하면? - a태그 안에 있는 엘리먼트를 클릭하면 ,, i는 아이콘 태그

p태그는 분단 태그) a태그 안에있는 온클릭 속성이 실행되는 것 !!

- pageSubmitFn() 는 호출함수(메서드)라고 부름, 스택에 이 메서드에 주소값이 쌓임

이 함수를 호출하고 있음.

<script>

**var** pageSubmitFn = **function**() {

alert("탄다!");

}

</script>

- 소괄호 앞에 함수명이 없으면 익명함수라고 한다.

익명함수를 지금 pageSubnitFn 의 변수에 넣었음, 이것은 변수인가 함수인가? 함수가 된다 !!

이렇게 넣는 순간 이 변수의 이름을 가진 함수가 된다.

location.href = "main.do";

- 레프트 메뉴를 누르면 메인.do로 가게 하고 싶음

지금 메인.do로 가게 하는 방법을 두가지를 배웠는데

첫번째 시간에 jsp:forward 방법으로 // 이번시간에 a herf 속성으로 하는 방법으로 두가지배움

- location.href = “main.do”; 로 보내면

get방식은 위에 주소에 해당 주소가 보이게 되는 되는 방식

- 호출함수 안에 내용이 main.do로 가라고 되어 있어서 dashboard2를 만들어도 main.do로 가고있음 이를 main2.do로 가기 위해서는 어떻게 해야 하는가? 구분할 수 있는 구분 값이 필요하다.함수 안에 인수, 인자를 넣었음 main이라는 문자열을 넣었음 근데 큰따음표 안에 작은따음표가 표준

- function(menu) 메뉴라는 매개변수안에는 main과 main2라는 값을 가짐

<script>

**var** name = "hanq";

**var** tempFn = **function**() {

**var** ban = "초급반";

alert("name1: " +name);

}

alert("name2: " + name);

tempFn();

</script>

**여기서 호출되는 순서는 name2 가 먼저 설정됨 // 여기서 ban 은 지역변수로 스코프 밖에서는 사용할 수 없음. 지역변수의 개념과 전역변수(필드, 멤버변수의 개념)를 알고 있어야 한다는 부분**

**var** pageSubmitFn = **function**(menu) {

**if** (menu === "main"){

location.href = "main.do?pageName=main"

} **else** **if** (menu === "main2") {

location.href = "main2.do?pageName=main2"

}

}

- pageName=main : 여기서 페이지 네임을 변수라고 생각하고 페이지 네임이라는 변수명에 메인이라는 문자열 값을 넣어서 메인.do로 보냈는데 이는 메인컨트롤러로 보내는 것

페이지 네임이라는 키에 담아서 보낸거 정확하게는 pageName 은 키 // main 은 밸류

?는 ?를 기준으로 좌측이 논리적 주소 // 우측이 데이터

키와 벨류 부분을 쿼리스트링 이라고 부를 것 (외울 것)

- === 값을 비교할때는 세번을 넣어야 표준임.

- if 문 for 문은 쓸 때 위를 한칸 비운다.

<li id=*"****main****"* class=*"menu active"*>

<a href=*"#"* onclick="pageSubmitFn('**main**')">

<i class=*"pe-7s-graph"*></i>

<p>Dashboard</p>

- 레프트의 흰색 바탕인 부분을 하이라이트 처리를 하여 현재 페이지가 어떤 페이지인지 구분하기 위해서 다시 설정할 예정

- 하이라이트 처리라는 것 자체가 클래스에서 처리를 해주는 것, f12를 눌러서 active클래스를 찾아야함 이 클래스는 li태그의 클래스임

- 우리가 지금 하고 싶은건내가 클릭한 화면만 엑티브를 남겨주고 나머지는 삭제해주면 됨

- 필요한 데이터와 필요한 작업을 확인해야함

- **호출함수를 호출할 때 인수로 내가 클릭한 클릭한 li의 id를 넘겨준다고 할 수 있는가? 있다.**

여기서 “main”을 얘기하는 것

@RequestMapping(value = "main.do")

**public** String initMain(HttpServletRequest request) **throws** Exception {

- HttpServletRequest request 는 화면에서 올라온 데이터가 쿼리스트링 방식이면 request라는 매개변수 안에 키-벨류 형태로 쌓인다. 쿼리스트링 방식이 아니면 쌓이지 않는다.

- 최초에 null 값이 나타나게 되는 이유가 구동순서를 알면 나오는데

서버를 껐다 키면 맨처음 호출되는 jsp는 index.jsp인데 여기에는 따로 pageName키를 보내지 않았기 때문에 처음에 null이 발생하는 것.

- 여태 키를 한 개를 보냈는데 두개를 보내기 위해서는 &(엠퍼센트)를 붙여서 보낸다.

겟방식으로 보낼 때 한글처리를 따로 하지 않으면 한글이 깨짐 //

이건 로컬 웹서버에서 해줘야 하는 것, 어? 우리 index.jsp에서 했잖아요? 이건 화면쪽 다국어

따라서

서버의 서버.xml에 들어가서 한글처리 해줘야함 65번째 라인에 추가함

<script>

$("#btn").click(**function**() {

alert("버튼을 클릭하셨습니다");

});

</script>

<button id = *"btn"*>버튼</button>

- 여기서 $ 는 쿼리

# 은 id

-따라서 $("#btn")는 <button id = *"btn"*>버튼</button> 을 말한다.

해당 타겟을 클릭하면 익명함수 안을 실행하겠습니다. 왜 익명함수? 어차피 이름이 필요가 없음

어차피 버튼만 클릭해도 제이쿼리 때문에 계속해서 실행되기 때문에 -----**클릭 이벤트 리스너 !**

. 을 기준으로 오른쪽은 메서드 왼쪽은 클래스

여기서는 클릭이라는 클릭이라는 메서드를 실행하고 있고 제이쿼리라는 클래스를 실행하고 있는것임.

근데 제이쿼리라는 클래스는 괄호안에 문자열 인수를 하나 넣었는데 제이쿼리는 클래스의 생성자를 이용하고 있는 것, 모든 자바클래스의 디폴트 생성자는 없음

클릭이라는 메서드는 제이쿼리의 것임

- 제이쿼리에 대한 설명을 몰아서 하지 않으므로 조금씩조금씩 기억하며 외울 것

오늘은 제이쿼리 클래스의 클릭 메서드를 배웠음 이를 외울 것

<script>

$(**function**(){

$("#btn").click(**function**() {

alert("버튼을 클릭하셨습니다");

})

})

</script>

<button id = *"btn"*>버튼</button>

- 도큐먼트레디 : 문서가 준비되면 내 안의 것을 실행해라

$(function(){

})

여기서의 문서는 html을 말하는 것

얘가 실행되는 것은 html이 실행되고 끝나면 실행되는 것 따라서 모든 jsp파일을 순서대로 하기 때문에 버튼과 순서를 상관없이

이벤트 리스너는 도큐먼트레디 안에서만 쓴다고 기억할 것 밖에서 쓰게되면 계산할게 많아짐

- 기본흐름은 위에서 아래!!

<script>

console.log("defaul-footer.jsp")

</script>

를 통해서 크롬 개발자 도구를 통해 순서를 파악할 수 있음

<li id=*"main"* class=*"menu active"*>

<a href=*"#"* onclick="pageSubmitFn('main')">

<i class=*"pe-7s-graph"*></i>

<p>Dashboard</p>

- 하이라이트를 다 지우기 위해서는 타겟을 잡아야 함

여기서 ‘다’ 는 li의 태그’들을’ 잡아야 함

- 여기서 클래스는 li태그는 두개의 클래스를 가지고 있는데 menu와 active 두개를 가지고 있음

- param : 이전 화면에서 보낸 키 를 얘기한다 (이를 el 태그)

이전화면에서 보낸 키를 적으면 페이지네임안에 들어가 있는 값을 넣어준다. 여기서 레프트의 첫번째를 누르면 메인이라는 문자가 담겨져 있음.

- 결국은 내가 보내서 내가 받았다는 이야기 ( 보낸 행위 location..herf= “main.do?pageName)

그치만 타일즈 구조상 우선순위가 레프트가 먼저기 때문에 param. 으로 메인으로 가기전에 디폴트 레프트가 받아내서 메인으로 가기전에 레프트로 받아와 실행이 가능한 것

- jstl을 쓸 예정 (파람을 c 태그로 감싸지 않으면 보안에 큰 문제가 생김

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%> 을 레프트에 추가함

var pageName = "<c:out value='${param.pageName}' />"; 에서

<c:out value='${ ~ }’ /> 을 사용하면 안의 물결모양의 내용이 문자열로 받아짐

여기서 “c”가 있는건 이 화면에서 jstl을 “c”로 쓰겠다고 지정해 주는 것

jstl을 쓰는 이유는 ? 66쪽을 그대로 실행하면

사용자가 홈페이지 관리자가 아니라도 정보를 긁어올 수 있는 함수가 실행 될 수있음

jstl을 쓰면 텍스트로 읽혀지게 만들어줌 함수로 읽는게 아니라

- 변수를 중간에 적으면 메모리 낭비가 이뤄지니 변수를 위로 올리는 코딩을 꼭 해야함

자동으로 올라가는 것은 호이스팅이라고 한다.

하이라이트는 어떻게 하는 거에요?

1. left에 있는 메뉴를 클릭한다는 것은 <li> 태그속 <a>태그를 누르는 것으로 해당 태그를 누를때 특정값을 인자값으로 보내기 위하여 onclick 이벤트로 pageSubmitFn 함수를 호출합니다.

2. 이 pageSubmintFn 호출함수 안에는 인자가 있으며, 이 인자는 해당 <a>태그가 속해있는 부모의 <li>태그의 id속성에 넣어진 value를 보냅니다.

3. pageSubmitFn은 function()라는 익명함수를 호출하는 변수로 pageSubmitFn 이라는 함수가 된다.

4. 아래의 oclick을 통해 들어온 value를 pageSubmitFn의 매개 변수로 받는다.

5. if 구문을 통해 매개변수의 값에 따라 메뉴를 눌렀을 때 가고자 하는 주소를 get방식으로 보내기 위하여 location.href를 이용

6. "main.do?pageName=main" 에서 '?'를 기준으로 왼쪽인 main.do로 가라는 논리적 주소와 '?'를 기준으로 오른쪽에서 데이터를 부여하는데 pageName=main 에서 pageName은 key, main은 value로 쌍을 이루는 쿼리스트링 방식으로 controller 에 보내준다.

7. default-left에서 보내진 'main.do'가 @RequestMapping의 값과 일치하여 해당 메서드안으로 들어감

8. HttpServletRequest request 은 화면에서 올라온 데이터가 쿼리스트링 방식이면 request라는 매개변수 안에 key-value방식으로 쌓인다.

9. request.getParameter(올린 key)를 적으면 해당 key에 있는 value값이 빠지고 리턴값에 main/main.tiles로 가라고 써있었으나 main화면으로 가지 않고 default-left가 타일즈 구조상 main보다 위에 있기 때문에 left가 먼저 가져와서 param을 이용하여 사용하게 됨.

10. param은 이전에 보낸 key를 입력하면 그 key와 대응되는 value의 값을 가져오게 한다. 따라서 첫번째 메뉴를 눌렀다고 가정했을대 param.pageName은 main이라는 value 값을 불러온다.

11. param.pageName를 var pageName으로 정의함으로써 위의 main이 pageName이 된다.

12. param을 쓰게 되면 보안상의 문제가 발생하기 때문에 jstl을 사용한다.

13. jstl을 이용하기 위하여 입력한다. 여기서 "c"는 앞으로 jstl을 "c"로 부름을 의미한다.

14. c:out value='${~}' /는 ~ 안의 내용을 문자열로 받는다.

15. $(function()){ } 는 jqery의 메서드로 document ready로 html이 실행되고 끝나고 난 뒤 실행된다.

16. document ready안에 $(".menu").removeClass("active") 를 입력하여 모든 menu class의 "acrive"를 삭제하여 하이라이트 표시를 없애준다

17. 이후 $("#" + pageName).addClass("active"); 를 입력하여 left menu에서 누른 <a>태그의 onclick 이벤트를 통하여 pageSubmitFn의 매개변수로 간 <li>의 id값이 document ready안에서 pageName으로 지정이되고 #pageName 즉, #main이 있는 <li>태그에 "acrive" class가 추가되며 하이라이트를 표시하게 된다.