

JSP와 MySQL을 이용한 제주도 호텔 예약 웹사이트 구현

김채연

프로젝트 명	JSP와 MySQL을 이용한 제주도 호텔 예약 웹사이트 구현
참여 인원	5명
담당 업무	호텔 객실 예약 페이지, 예약 내역 확인 페이지 구현
구현	1. 객실 예약 기능 구현 2. 로그인한 유저의 예약 내역을 확인할 수 있는 페이지 구현 3. 예약 내역 확인 페이지에서 예약 수정, 취소할 수 있는 기능 구현
개발 환경	Windows10
사용 도구	Eclipse, Dbeaver, MySQL, JDBC, Apache Tomcat, Draw.io, Object Aid, Slack, Figma
사용 기술	Java, JDBC, JavaScript, jQuery, HTML, CSS, JSP, SQL



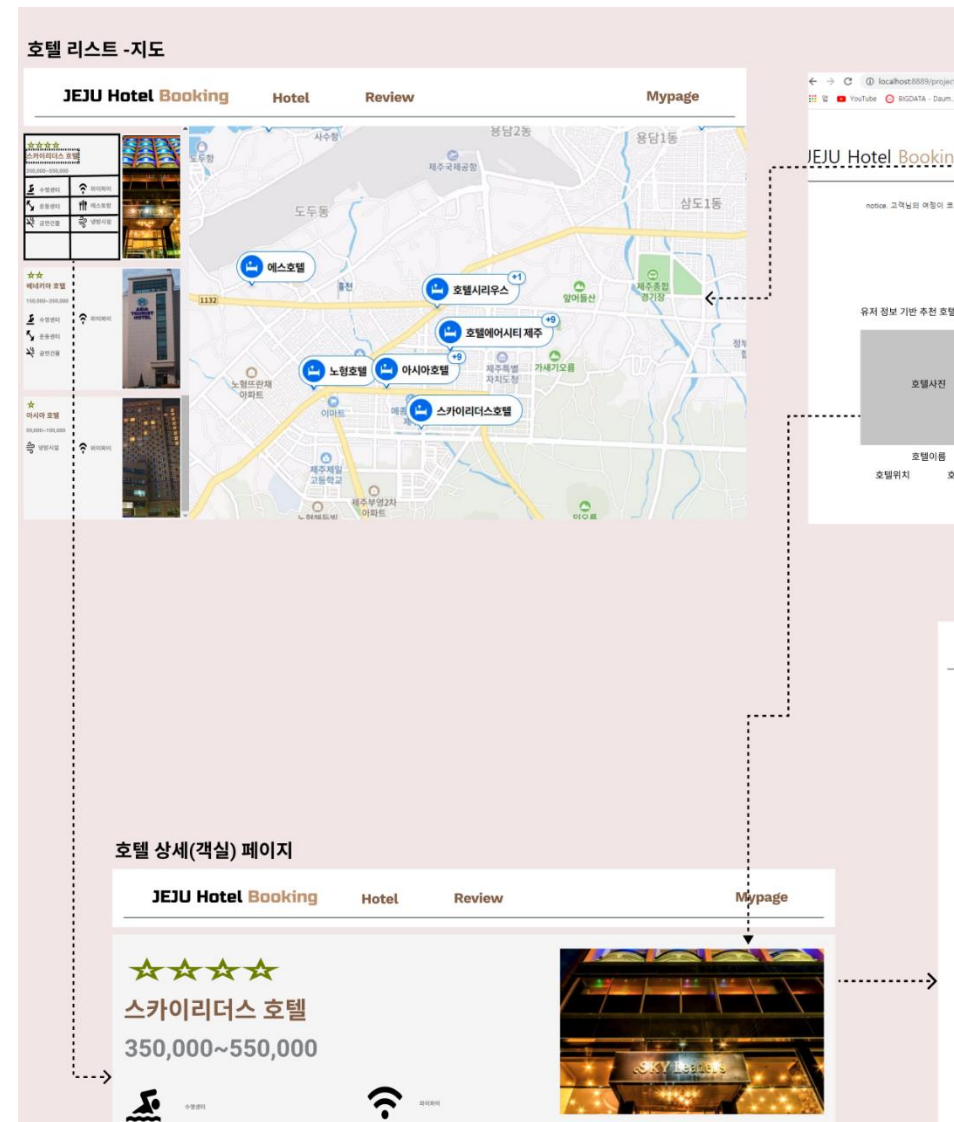
The image displays two screenshots of the Slack web interface, illustrating a workspace for a project named '3-1 안드로이드 검색'.

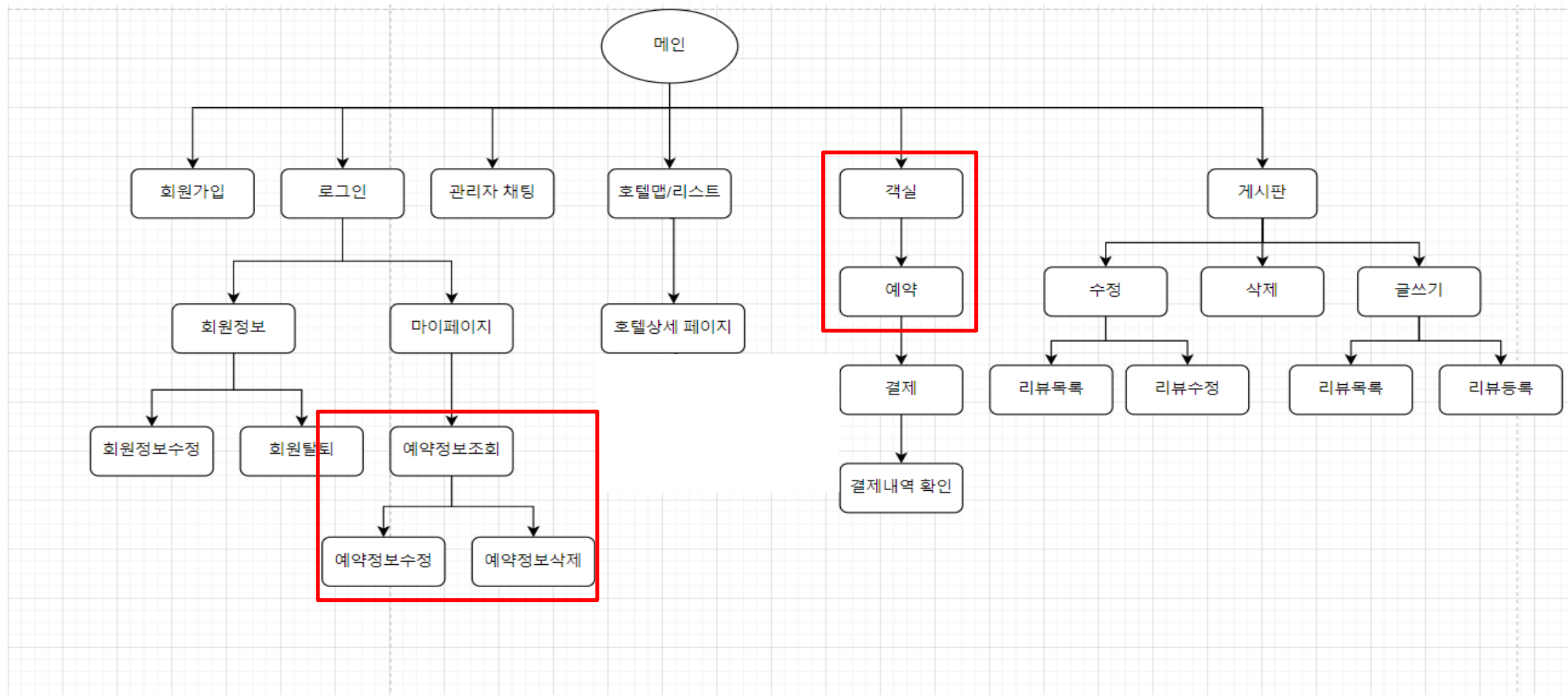
Left Screenshot: Channel #개인-김채연

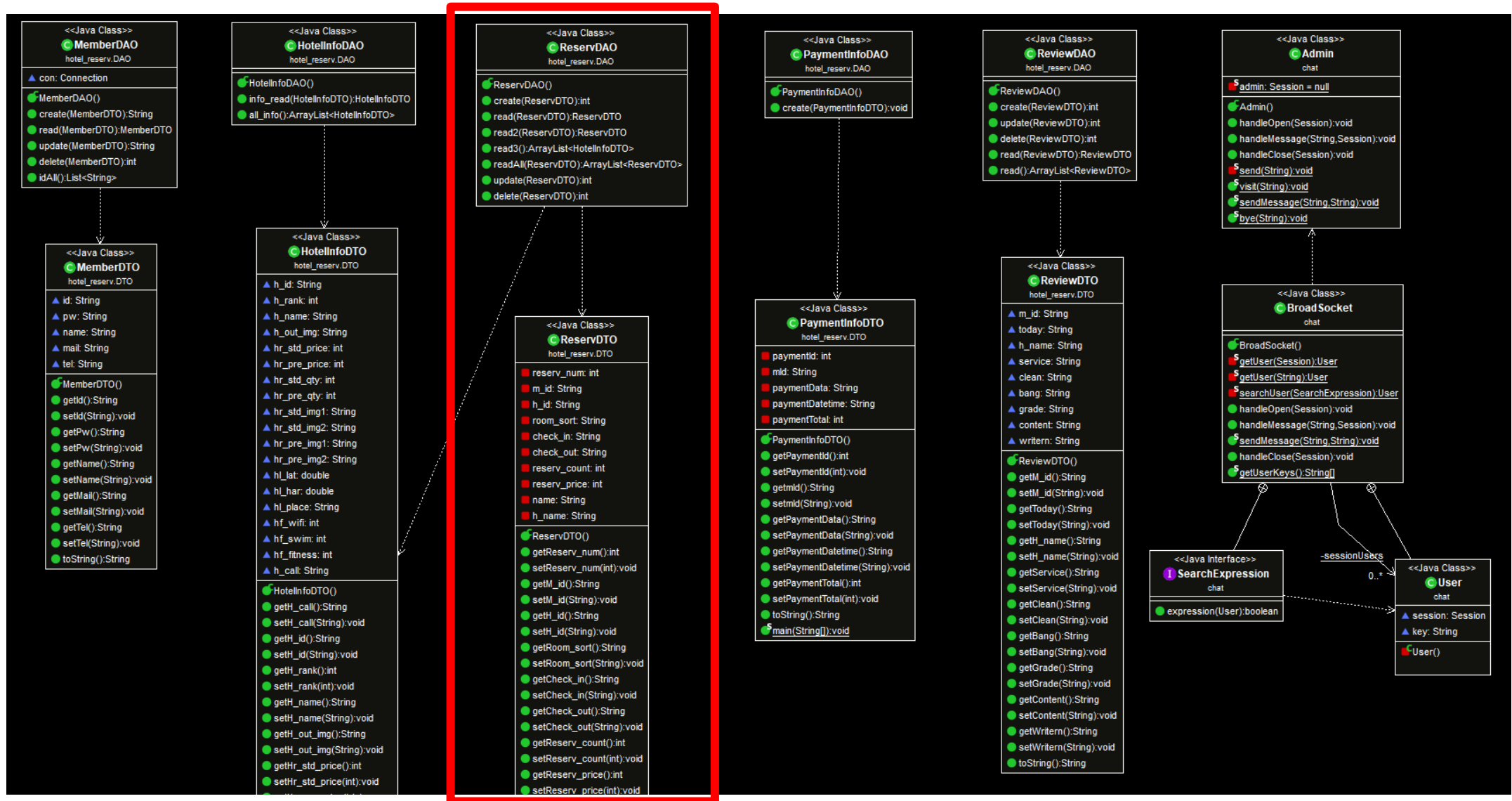
- Channel Name:** #개인-김채연
- Message from 김채연 (오전 11:54):**
 - <변경사항>
 - DB 모든 컬럼 nn체크
 - <합할 때 꼭 넣어야 보이는 파일>
 - reserv_image.css
 - img 폴더 내 이미지들
 - 3개 파일
 - image.png (106kB PNG)
- Message from 김채연 (오전 9:43):**
 - 최종
 - Zip
 - project01_reservation.zip (18kB Zip)

Right Screenshot: Channel #컨퍼런스

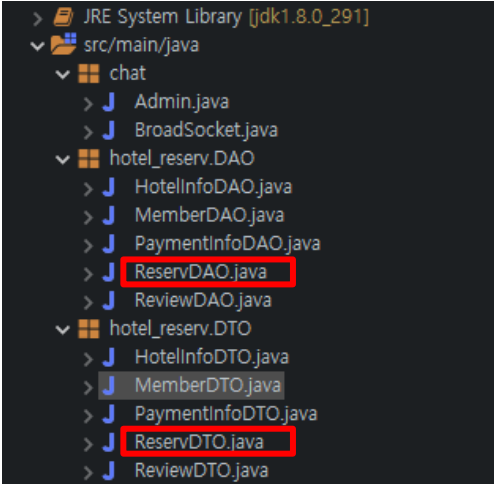
- Message from 김채연 (오후 1:22):**
 - DB명: reservation (숙소 예약)
 - 컬럼:
 1. reserNum(예약번호)
 2. reservId(예약자명)
 3. reservHotel(호텔명)
 4. roomSort(객실종류)
 5. roomNum(객실번호)
 6. checkin(입실날짜)
 7. checkout(퇴실날짜)
 8. reservCount(숙박인원)
 9. reservPrice(금액)
- Message from 변승원 (오후 5:07):**
 - 장바구니 페이지에서 사용될 DB
 - read (from 호텔DB 명준씨) 상품id, 상품명, 상품가격
 - create 장바구니id(pk), 사용자id, 상품id, 수량, 총금액
 - 결제 페이지에서 사용될 DB
 - read (from 예약DB 채연씨) 총금액
 - create 결제id(pk), 결제금액, 결제일, 결제자성명, 결제회사 (편집됨)
- Message from 이준규 (오후 5:11):**
 - image.png
- Message from 양샘이 (오후 5:12):**
 - #컨퍼런스에게 메시지 보내기







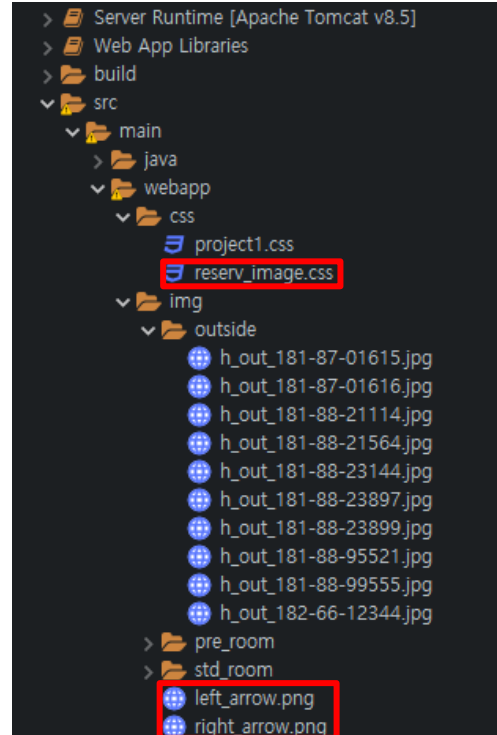
Back-end



JAVA

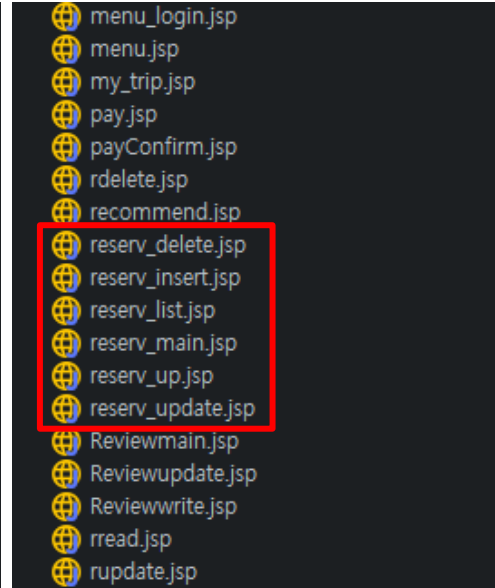
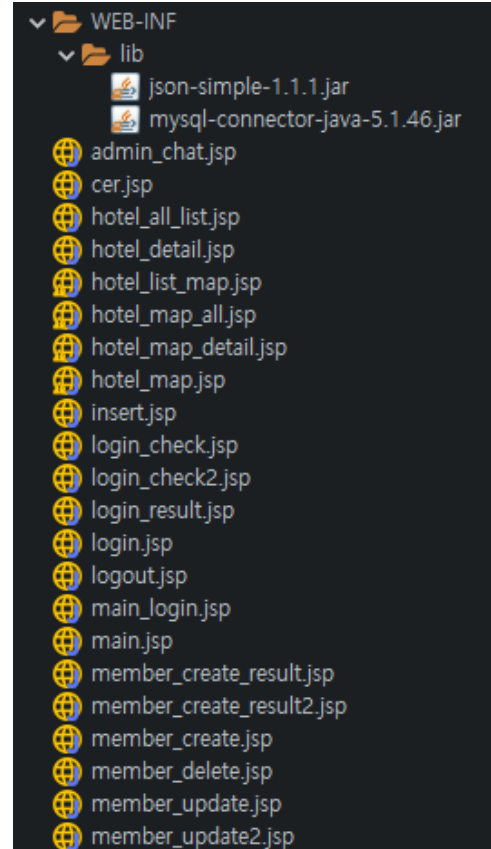
- ReservDAO.java
- ReservDTO.java

Front-end



CSS

- reserve_image.css



JSP

- reserve_main (예약 메인페이지)
- Reserv_list (예약내역 페이지)
- Reserv_up (예약내역 상세페이지)
- reserv_insert (예약 등록)
- Reserv_update (예약 수정)
- Reserv_delete (예약 취소)

Table

3조		작성자: 김채연						
DB	Table	Num	Column	type	Auto Increment	NN	KEY	Comment
hotel_reserv	reservation	1	reserv_num	int	O	O	PK	예약자_예약번호
		2	m_id	varchar(50)		O		예약자_아이디
		3	name	varchar(20)		O		예약자_이름
		4	h_id	char(12)		O		기본_호텔아이디
		5	h_name	varchar(50)		O		호텔_이름
		6	room_sort	varchar(20)		O		객실_종류
		7	check_in	varchar(20)		O		입실_날짜
		8	check_out	varchar(20)		O		퇴실_날짜
		9	reserv_count	int		O		숙박_인원
		10	reserv_price	int		O		예약_금액

ERD

 reservation

 reserv_num

ABC m_id

ABC name

ABC h_id

ABC h_name

ABC room_sort

ABC check_in

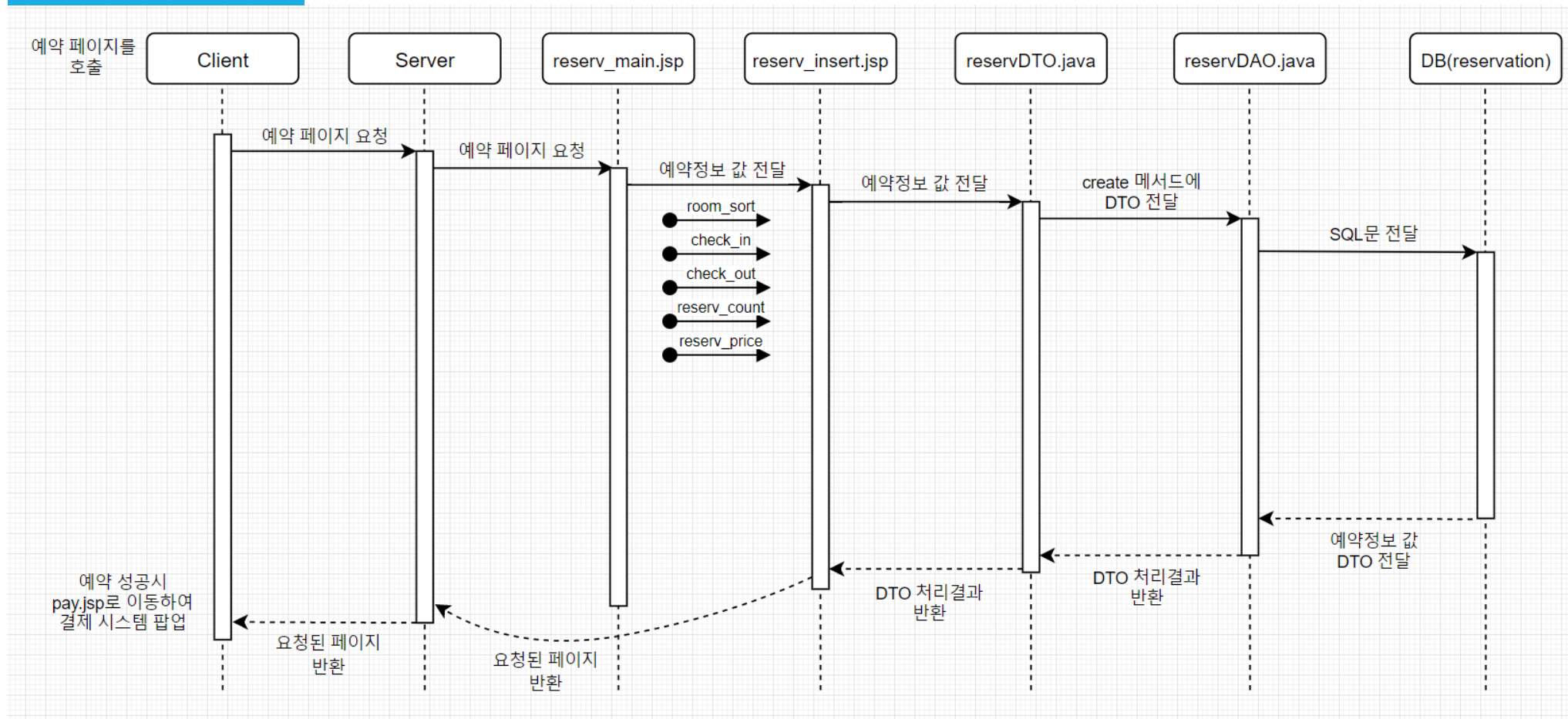
ABC check_out

123 reserv_count

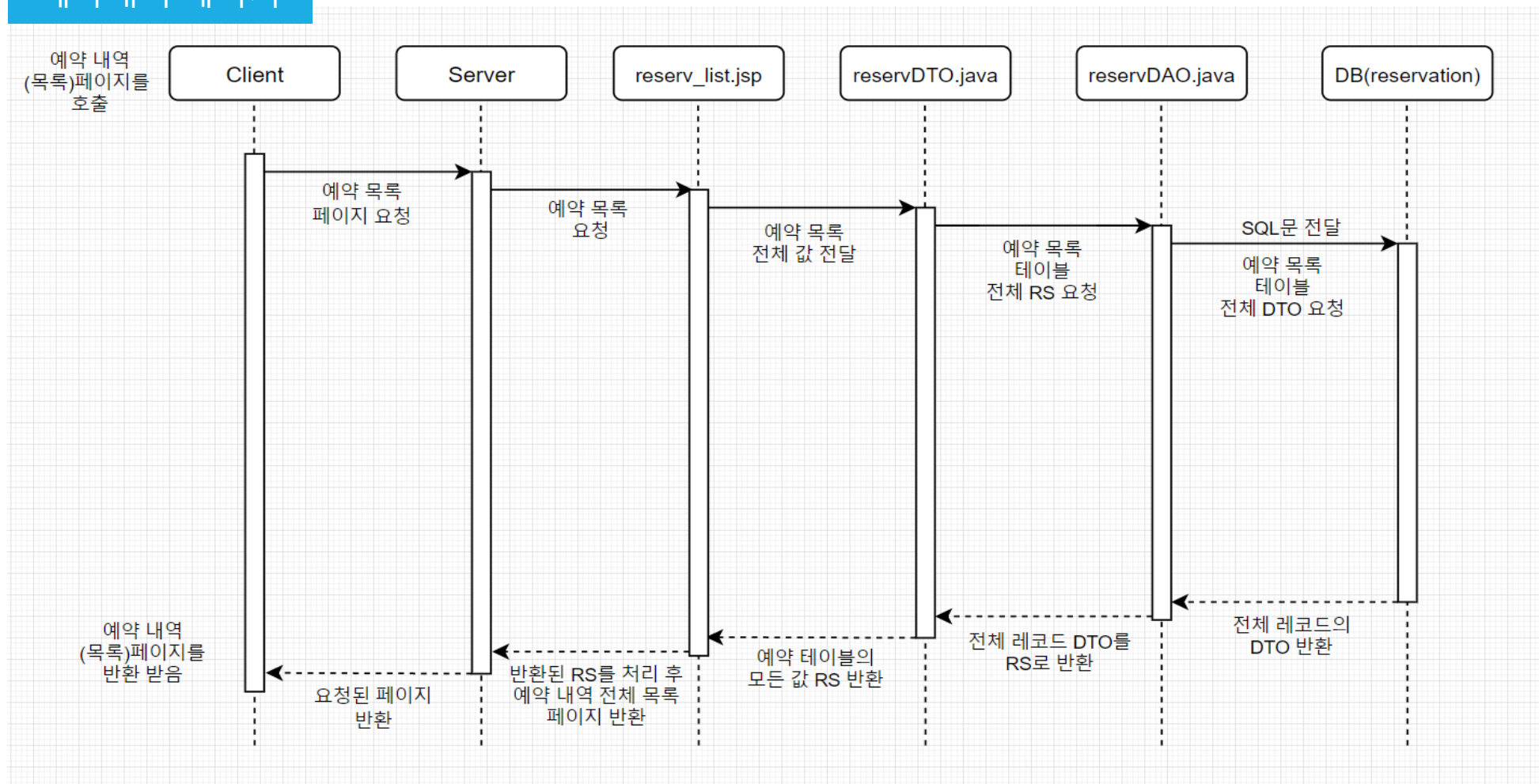
123 reserv_price

Sequence Diagram

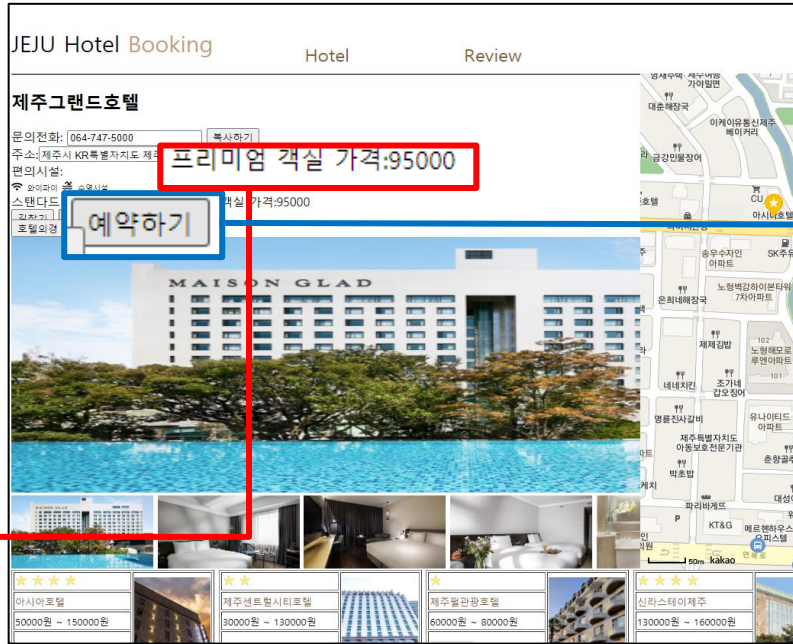
예약 메인페이지



예약내역 페이지



- UI



호텔 상세정보에서
예약 페이지로 이동

세션 받은
값을 전달

```
//예약페이지 이미지 정보 세션으로 받아올 값
String hr_std_img1 = (String)session.getAttribute("hr_std_img1");
String hr_std_img2 = (String)session.getAttribute("hr_std_img2");
String hr_pre_img1 = (String)session.getAttribute("hr_pre_img1");
String hr_pre_img2 = (String)session.getAttribute("hr_pre_img2");

//예약자 정보 세션으로 받아올 값 4개 (name, h_name, h_std_price, h_pre_price)
String name = (String)session.getAttribute("name"); //준규DB 예약자명
String h_name = (String)session.getAttribute("h_name"); //명준DB 호텔명
int h_std_price = (int)session.getAttribute("hr_std_price"); //명준DB std가격
int h_pre_price = (int)session.getAttribute("hr_pre_price"); //명준DB pre가격
```

notice. 고객님의 여정이 코로나19(COVID-19)발생 상황에 영향을 받을 수 있으니 예약시 코로나 협참을 꼭 확인해주세요.



예약자 정보

예약자명	strawberry	<%= name %>
호텔명	제주그랜드호텔	<%= h_name %>

객실 정보

객실 타입	프리미엄
입실 일자	2021-08-13
퇴실 일자	2021-08-15
투숙 인원	3명 (기준 2인/최대 3인, 기준 인원 초과시 추가비용 발생. 스탠다드-25000원, 프리미엄-70000원)
결제 금액	95000원 (- 70000원) x 2일 = 총금액: 330000원

결제하기

취소하기

<%= h_pre_price %>

- 객실 내부 이미지(Slide)

reserv_image.css



```

<!-- 객실 내부 이미지(Slide) -->
<div class="section">
  <input type="radio" name="slide" id="slide01" checked="" />
  <input type="radio" name="slide" id="slide02" />
  <input type="radio" name="slide" id="slide03" />
  <input type="radio" name="slide" id="slide04" />
  <div class="slidewrap">
    <ul class="slidelist">
      <li>
        <a> <!-- a태그로 배너처럼 클릭해 링크이동 가능 -->
          <label for="slide04" class="left"></label>
           <!-- 이미지 세션값 -->
          <label for="slide02" class="right"></label>
        </a>
      </li>
      <li>
        <label for="slide01" class="left"></label>
        
        <label for="slide03" class="right"></label>
      </li>
      <li>
        <label for="slide02" class="left"></label>
        
        <label for="slide04" class="right"></label>
      </li>
      <li>
        <label for="slide03" class="left"></label>
        
        <label for="slide01" class="right"></label>
      </li>
    </ul>
  </div>
</div><br>

```

초기 이미지

속성값: 이전 이미지

속성값: 다음 이미지

left 클래스: < 버튼

right 클래스: > 버튼

첫 번째 스크린

두 번째 스크린

세 번째 스크린

네 번째 스크린

radio에 동일한 name값을 주어 그룹핑한다(1개만 체크되는 효과)

```

@charset "UTF-8";
/* 어딘가 안 맞을 땐 F12-select an element로
   요소들 값 확인하기 */
.section input[id*="slide"]{
  display:none;
}
.section .slidewrap{
  max-width:800px;
  margin: 0;
  overflow: hidden;
}
.section .slidelist{
  white-space: nowrap;
  font-size: 0;
  padding-left: 0; /* ul의 왼쪽 padding을 없애줘야 위치도 좌표도 맞음!!! */
}
.section .slidelist > li{
  display: inline-block;
  vertical-align: middle;
  width: 100%;
  transition: all .5s;
}
.section .slidelist > li > a{
  display: block;
  position: relative;
}
.section .slidelist > li > a img{
  width: 800px;
  height: 388px;
}
.section .slidelist label{
  position: absolute;
  z-index: 10;
  top: 50%;
  transform: translateY(-50%);
  padding: 50px;
  cursor: pointer;
}

```

section 클래스 안의, id값에 "slide"를 가진(*) input을 선택

화면에서 안 보이게 설정

영역을 벗어나면 안 보이게

Space, tab: 병합

줄바꿈: 병합

자동 줄바꿈: x

자연스러운 애니메이션 효과

속성값: 전체 옵션(all), .5s: 슬라이드의 빠르기/시간 조정

안쪽에 콘텐츠를 다 담을 수 있도록

버튼으로 쓸 <label>을 자을 배치

어느 객체가 앞으로 나오고, 뒤에 나올지 순서 결정

위로부터 50%인 지점에 위치

자신을 기준으로 위로 50% 당기기

마우스를 올렸을 때 pointer로 커서 변경

- 객실 내부 이미지(Slide)

reserv_image.css



라벨 클릭 효과

```
.section .slidelist .left{
  left: 10px;
  background: url('../img/left_arrow.png') center center / 100% no-repeat;
}
.section .slidelist .right{
  right: 10px;
  background: url('../img/right_arrow.png') center center / 100% no-repeat;
}
.section [id="slide01"]:checked ~ .slidewrap .slidelist > li {transform:translateX(0%);}
.section [id="slide02"]:checked ~ .slidewrap .slidelist > li {transform:translateX(-100%);}
.section [id="slide03"]:checked ~ .slidewrap .slidelist > li {transform:translateX(-200%);}
.section [id="slide04"]:checked ~ .slidewrap .slidelist > li {transform:translateX(-300%);}
```

첫번째 input이 체크가 되면,

Input아래의 slidewrap 클래스 자식 요소인 li에 style 주기 자신을 기준으로 x좌표 0값 안에 있는 slidelist 클래스

- 선택 항목에 따라 항목 값 초기화

세션값

입력값

예약자 정보	
예약자명	strawberry
호텔명	제주그랜드호텔

객실 정보	
객실 타입	프리미엄
입실 일자	2021-08-13
퇴실 일자	2021-08-15
투숙 인원	3명
결제 금액	95000원 (+70000원) x 2일 = 총금액: 330000원

결제하기 취소하기

각 항목의 초기값 (String)

#sortId	r_default
#inId	yyyy-mm-dd
#outId	yyyy-mm-dd
#countId	0
#pString	0원
#pricId	-

날짜 일수를
계산해주는 함수 dayDiff()

```
//날짜 일수 계산
function dayDiff() {
    date1 = $('#inId').val() //yyyy-mm-dd, String. 작은 수
    date2 = $('#outId').val() //큰 수
    arr1 = date1.split('-')
    arr2 = date2.split('-')
    d1 = new Date(arr1[0], arr1[1], arr1[2]) //Date객체로 변환
    d2 = new Date(arr2[0], arr2[1], arr2[2])
    diff = d2 - d1 //날짜 차이 알아내기
    currDay = 24 * 60 * 60 * 1000 //시*분*초*밀리세컨
    DateDiff = parseInt(diff / currDay) //일수 차이
}
```

```
$(function() {
    $('#sortId').change(function() {
        //객실타입이 바뀌면 입퇴실 일자, 인원, 금액 초기화
        $('#countId').val('0');
        refactoring();
        $('#inId').val('yyyy-mm-dd');
        $('#outId').val('yyyy-mm-dd');
        //퇴실 일자 변경 시 인원, 금액 초기화
        $('#outId').change(function() {
            $('#countId').val('0');
            refactoring();
            //3일 이상 예약 불가 처리
            dayDiff();
            if(DateDiff > 3){
                alert('최대 3일까지만 예약할 수 있습니다.')
                $('#countId').val('0');
                refactoring();
                $('#outId').val('yyyy-mm-dd');
            }
            if(date1 > date2 || date1 == date2){
                alert('입실/퇴실 일자를 확인해주세요.')
                $('#outId').val('yyyy-mm-dd')
            }
        })
    })
})
```

<가격을 초기화해야 하는 때>

1. 객실 타입이 바뀔 때
2. 투숙 기간이 바뀔 때
3. 투숙 인원이 3명일 때

```
//중복코드(가격 초기화) 정리
function refactoring() {
    $('#pString').val('0원');
    $('#priceId').val('');
}
```

2021년 08월

일	월	화	수	목	금	토
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28

localhost:8887 내용:

최대 3일까지만 예약할 수 있습니다.

확인

- 결제 금액 계산

세션값

입력값

예약자 정보	
예약자명	strawberry 2
호텔명	제주그랜드호텔

객실 정보	
객실 타입	프리미엄
입실 일자	2021-08-13
퇴실 일자	2021-08-15
투숙 인원	3명 (기준 2)
결제 금액	95000원 (+70000원) x

결제하기 취소하기

각 항목의 초기값 (String)

#sortId	r_default
#inId	yyyy-mm-dd
#outId	yyyy-mm-dd
#countId	0
#pString	0원
#priceId	-

날짜 일수를
계산해주는 함수 dayDiff()

```

$('#countId').change(function() {
    //중복코드(가격 초기화) 정리 목적 함수 refactoring()
    refactoring()
    //입퇴실 일자 간의 차이가 0보다 크면 뺀 일수를 가격에 곱해주기
    //result1은 총금액, result2는 금액 계산과정 (금액*일수=총금액)
    //날짜 일수 계산 함수 dayDiff()
    dayDiff()
    if($('#sortId').val() == "스탠다드" && DateDiff > 0 && ($('#countId').val() == "1" || ($('#countId').val() == "2")){
        result1 = (<%= h_std_price %> * DateDiff) + '원'
        result2 = <%= h_std_price %> + '원 ' + 'x' + DateDiff + '일 = 총금액 : '
        $('#pString').val(result2); $('#priceId').val(result1);
    }else if($('#sortId').val() == "스탠다드" && DateDiff > 0 && ($('#countId').val() == "3")){
        result1 = ((<%= h_std_price %> + 25000) * DateDiff) + '원'
        result2 = <%= h_std_price %> + '원 (+25000원)' + 'x' + DateDiff + '일 = 총금액 : '
        $('#pString').val(result2); $('#priceId').val(result1);
    }
    if($('#sortId').val() == "프리미엄" && DateDiff > 0 && ($('#countId').val() == "1" || ($('#countId').val() == "2")){
        result1 = (<%= h_pre_price %> * DateDiff) + '원'
        result2 = <%= h_pre_price %> + '원 ' + 'x' + DateDiff + '일 = 총금액 : '
        $('#pString').val(result2); $('#priceId').val(result1);
    }else if($('#sortId').val() == "프리미엄" && DateDiff > 0 && ($('#countId').val() == "3")){
        result1 = ((<%= h_pre_price %> + 70000) * DateDiff) + '원'
        result2 = <%= h_pre_price %> + '원 (+70000원)' + 'x' + DateDiff + '일 = 총금액 : '
        $('#pString').val(result2); $('#priceId').val(result1);
    }
})//#countId change
  
```

결제 금액

70000원 (+25000원) x 2일 = 총금액 :

190000원

결제/취소 버튼 기능

1개 이상의 항목이
기본(디폴트)값일 경우

2

예약자 정보	
예약자명	strawberry
호텔명	제주그랜드호텔

객실 정보	
객실 타입	--객실선택--> "r_default"
입실 일자	연도-월-일 <input type="text" value="" /> " "
퇴실 일자	연도-월-일 <input type="text" value="" /> " "
투숙 인원	--인원선택--> "0" 최대 3인, 기준 인원 초과시 추가비용 발생. 스탠다드-25000원, 프리미엄-70000원
결제 금액	0원

localhost:8887 내용:
모든 항목에 응답해주세요.

확인

check() 함수

- 유효성 체크
: 입력하지 않은 값이 있는지 체크
- 입실 일자가 퇴실 일자보다
먼 날짜로 설정 못하도록 체크

```
//유효성, 입실일자>퇴실일자 체크
function check() {
    if($('#sortId').val()=="r_default" || $('#inId').val()==" " || $('#outId').val()==" " || $('#countId').val()=="0"){
        alert('모든 항목에 응답해주세요.')
        return false
    }
    date1 = $('#inId').val()
    date2 = $('#outId').val()
    if(date1 > date2 || date1 == date2){
        alert('입퇴실 일자를 확인해주세요.')
        return false
    }
}
```

→ return 값이 true면
form의 입력값을 전송,
false면 전송하지 않음

예약 진행 취소 시
호텔 상세페이지로 이동

```
<!-- check()로 유효성, 입실일자>퇴실일자 체크 후 리턴 값이 true일 때 폼 값 제출 -->
<form action="reserv_insert.jsp" onsubmit="return check()">
```

```
<input type="submit" value="결제하기" style="float:left;
margin-top: 10px; margin-right: 20px; width: 120px; height: 30px">
</form>
```

```
<!-- 취소하기 버튼 실행시 이전 페이지로 이동 -->
<a href="hotel_detail.jsp">
<button style="float:left; margin-top: 10px; margin-right: 20px;
width: 120px; height: 30px">취소하기</button></a>
```



- 결제 후 예약 정보가 목록에 들어옴

예약자 정보

예약자명	strawberry
호텔명	제주그랜드호텔

객실 정보

객실 타입	프리미엄
입실 일자	2021-08-13
퇴실 일자	2021-08-15
투숙 인원	3명 (기준 2인/최대 3인, 기준 인원 초과시 추가비용 발생)
결제 금액	95000원 (+70000원) x 2일 = 총금액 : 330000원

결제하기 취소하기

JEJU Hotel Booking

Hotel

Review

1

my trip

logout

reservDAO.java

```
public int create(ReservDTO dto) {
```

notice, 고객님의 여정이 코로나19(COVID-19)발생 상황에 영향을 받을 수 있으니 예약시 코로나 현황을 꼭 확인해주세요.

회원정보

2

예약정보

전체 예약된 객실의 개수: 2개

```
// 3. sql문을 만든다.
//Auto-increment는 insert할 때 생성됨(null을 넣어둠)
String sql = "insert into reservation values (null, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)";
PreparedStatement ps = con.prepareStatement(sql);
```

예약 번호	예약자명	호텔명	객실 타입	입실 일자	퇴실 일자	투숙 인원	결제 금액	
98	strawberry	제주센트럴시티호텔	스탠다드	2021-09-05	2021-09-06	1	30000	예약변경 예약취소
99	strawberry	제주그랜드호텔	프리미엄	2021-08-13	2021-08-15	3	330000	예약변경 예약취소

reserv_list.jsp

```
<!-- 좌측 메뉴바(회원정보, 예약정보) -->
</div>
<div id="my_trip">
  <ul>
    <li class='my_trip_button'><a href="my_trip.jsp">회원정보</a></li>
    <li class='my_trip_button'><a href="reserv_list.jsp">예약정보</a></li>
  </ul>
</div>
<div id="updown" style="margin-right:25px"></div>
```


- 동일한 ID의 예약 정보들을 읽어옴

전체 예약된 객실의 개수: 2개

예약 번호	예약자명	호텔명	객실 타입	예약자 정보	
98	strawberry	제주센트럴시티호텔	스탠다드	예약자명	strawberry
				호텔명	제주그랜드호텔
99	strawberry	제주그랜드호텔	프리미엄	2021-09-05	2021-09-06
				2021-08-13	2021-08-15

reserv_list.jsp

```
<%
//로그인한 id로 예약된 글들을 검색하기 위해 id값을 세션 받아옴
//메인-예약정보로 바로 들어온 것이라 예약페이지에서 잡아둔 세션은 이어지지 않고,
//새로 세션을 잡아줘야 한다.
String m_id = (String)session.getAttribute("id"); //예약자id(준규DB)
//유저의 id를 dto에 실는다.
ReservDTO dto = new ReservDTO();
dto.setM_id(m_id);
//1개 이상 검색하는 readAll메서드에 dto를 입력값으로 전송
ReservDAO dao = new ReservDAO();
ArrayList<ReservDTO> list = dao.readAll(dto);
//반환된 값은 list에, 동일한id값을 갖는 레코드가 담겨있다.
%>
```

reservDAO.java

```
// 1개 이상 검색하는 readAll / dto6
public ArrayList<ReservDTO> readAll(ReservDTO dto) { //dto전달받음

    // 3. sql문을 만든다.
    // 로그인한 id로 작성된 예약글 모두 검색
    String sql = "select * from reservation where m_id = ?";
    PreparedStatement ps = con.prepareStatement(sql);
    ps.setString(1, dto.getM_id());
    System.out.println("3. sql문 생성 성공!!!");

    // 4. sql문을 mysql로 전송한다.
    rs = ps.executeQuery();
    System.out.println("4. sql문 전송 성공");
    while (rs.next()) {
        //입력용dto, 반환용dto 따로 만들어야함!!! 재사용x
        ReservDTO dto6 = new ReservDTO(); //반환용dto6
        System.out.println("검색 결과가 있음.");
        dto6.setReserv_num(rs.getInt(1));
        dto6.setM_id(rs.getString(2));
        dto6.setName(rs.getString(3));
        dto6.setH_id(rs.getString(4));
        dto6.setH_name(rs.getString(5));
        dto6.setRoom_sort(rs.getString(6));
        dto6.setCheck_in(rs.getString(7));
        dto6.setCheck_out(rs.getString(8));
        dto6.setReserv_count(rs.getInt(9));
        dto6.setReserv_price(rs.getInt(10));
        System.out.println(dto6);
        list.add(dto6);
        System.out.println(list.size()); // 0개 나오는 애러 방지: 넘겨
    }
} catch (ClassNotFoundException e) { // 1단계
    System.out.println("1번 애러>> 드라이버 없음!!!!");
    e.printStackTrace(); // 애러를 구체적으로 알려줌
} catch (SQLException e) { // 2-4단계
    System.out.println("2-4번 애러>> DB관련된 처리하다 애러 발생!!!!");
    e.printStackTrace();
}

return list;
} // read end
```

- 동일한 ID의 예약 정보들을 읽어옴

전체 예약된 객실의 개수: 2개

예약 번호	예약자명	호텔명	객실 타입	입실 일자	퇴실 일자
98	strawberry	제주센트럴시티호텔	스탠다드	2021-09-05	2021-09-06
99	strawberry	제주그랜드호텔	프리미엄	2021-08-13	2021-08-15

reserv_list.jsp

```
<%
//로그인한 id로 예약된 글들을 검색하기 위해 id값을 세션 받아옴
//메인-예약정보로 바로 들어온 것이라 예약페이지에서 잡아둔 세션은 이어지지 않고,
//새로 세션을 잡아줘야 한다.
String m_id = (String)session.getAttribute("id"); //예약자id(준규DB)
//유저의 id를 dto에 싣는다.
ReservDTO dto = new ReservDTO();
dto.setM_id(m_id);
//1개 이상 검색하는 readAll메서드에 dto를 입력값으로 전송
ReservDAO dao = new ReservDAO();
ArrayList<ReservDTO> list = dao.readAll(dto);
//반환된 값은 list에, 동일한id값을 갖는 레코드들이 담겨있다.
%>
```

```
<br><br><b>전체 예약된 객실의 개수: <%= list.size() %>개</b><br>
<hr style="width: 220px; color: lightgray; margin-bottom: 10px">
<table border=1>
  <tr bgcolor="beige">
    <th width="90">예약 번호</th>
    <th width="100">예약자명</th>
    <th width="150">호텔명</th>
    <th width="100">객실 타입</th>
    <th width="100">입실 일자</th>
    <th width="100">퇴실 일자</th>
    <th width="90">투숙 인원</th>
    <th width="100">결제 금액</th>
    <th width="100"></th>
  </tr>
  <%
  for(ReservDTO dto6 : list){
    <tr align="center">
      <td><%= dto6.getReserv_num() %></td>
      <td><%= dto6.getName() %></td>
      <td><%= dto6.getH_name() %></td>
      <td><%= dto6.getRoom_sort() %></td>
      <td><%= dto6.getCheck_in() %></td>
      <td><%= dto6.getCheck_out() %></td>
      <td><%= dto6.getReserv_count() %></td>
      <td><%= dto6.getReserv_price() %></td>
      <td>
        <a href="reserv_up.jsp?reserv_num=<%=dto6.getReserv_num()%>">
          <button style="width:100px">예약변경</button></a>
        <a href="reserv_delete.jsp?reserv_num=<%=dto6.getReserv_num()%>">
          <button style="width:100px">예약취소</button></a>
        </td>
      </tr>
    <% } %>
  </table>
```

ID가 "strawberry"인 레코드들이 담겨있는 list. 반복문으로 출력

- UI

JEJU Hotel Booking

Hotel

Review

notice. 고객님의 여정이 코로나19(COVID-19)발생 상황에 영향을 받을 수 있으니 예약시 코로나 현황을 꼭 확인해주세요.

회원정보

예약정보

strawberry님의 예약 정보

예약 번호	98
예약자id	strawberry
예약자명	strawberry
호텔명	제주그랜드호텔
객실 타입	--객실선택-- ▾
입실 일자	연도 - 월 - 일 
퇴실 일자	연도 - 월 - 일 
투숙 인원	--인원선택-- ▾ (기준 2인/최대 3인, 기준 인원 초과시 추가비용 발생. 스탠다드-25000원, 프리미엄-70000원)
결제 금액	0원

예약 변경

변경 취소

- 예약번호로 특정 예약 정보 조회

전체 예약된 객실의 개수: 2개

reserv_list.jsp

`<a href="reserv_up.jsp?reserv_num=<%=dto6.getReserv_num()%>">`
`<button style="width:100px">예약변경</button>`

예약 번호	예약자명	호텔명	객실 타입	입실 일자	퇴실 일자	투숙 인원	결제 금액	예약변경
98	strawberry	제주센트럴시티호텔	스탠다드	2021-09-05	2021-09-06	1	30000	예약변경

reservDAO.java

JEJU

```
//1개 검색하는 read
public ReservDTO read(ReservDTO dto) {
    // 3. sql문을 만든다.
    String sql = "select * from reservation where reserv_num = ?";
}
```

회원정보

예약정보

strawberry님의 예약 정보

예약 번호	98
예약자id	strawberry
예약 호텔	예약 번호: 98
객실 타입	--객실선택--
입실 일자	연도-월-일
퇴실 일자	연도-월-일
투숙 인원	--인원선택-- (기준 2인/최대 3인, 기준 인원 초과시 추가비용 발생. 스탠다드-25000원, 프리미엄-70000원)
결제 금액	0원

예약 변경

변경 취소

반환받은 값 표현식으로 출력

객실 가격은 hotel_info DB에서 찾아온다.

테이블 내의
기능 구현
코드는
예약 메인
페이지와
동일

reserv_up.jsp

```
<%
//a태그로 받아온 예약번호 값으로 특정한 예약 내역의 상세페이지 조회하기
String reserv_num = request.getParameter("reserv_num");
int reserv_num2 = Integer.parseInt(reserv_num);
String m_id = (String)session.getAttribute("id"); //예약자id(from준규D
String name = (String)session.getAttribute("name"); //예약자명(from준규
//조건 걸 값 dto에 끼우기
//1개의 레코드를 검색하는 read메서드에 dto 실기
ReservDTO dto = new ReservDTO();
dto.setM_id(m_id);
dto.setReserv_num(reserv_num2);
//특정 예약번호의 레코드 1개를 읽어와 dto2에 실기
ReservDAO dao = new ReservDAO();
ReservDTO dto2 = dao.read(dto);
//dto2에서 필요한 값을 꺼내기
//여기까지는 reservation DB에 값이 있었지만,
String h_id = dto2.getH_id();
String h_name = dto2.getH_name();
//스탠다드, 프리미엄 객실 가격 값은 db에 없으므로
//hotel_info DB에서 찾아와야한다.
//HotelInfoDTO에 dto3를 만들고 dto2에서 꺼내온 h_id값을 실는다.
//(HotelInfoDAO의 info_read메서드는 h_id를 조건으로 레코드 1개를 검색)
HotelInfoDTO dto3 = new HotelInfoDTO();
dto3.setH_id(h_id);
//HotelInfoDAO의 info_read메서드에 dto3을 입력값으로 전송
//반환된 값(특정 h_id를 가진 레코드)을 dto4에 실기
HotelInfoDAO dao2 = new HotelInfoDAO();
HotelInfoDTO dto4 = dao2.info_read(dto3);
//dto4에서 필요한 객실 가격 값을 꺼내오기
int h_std_price = dto4.getHr_std_price();
int h_pre_price = dto4.getHr_pre_price();
%>
```

- 예약번호로 특정 예약 정보 변경

strawberry님의 예약 정보

예약 번호	98
예약자id	strawberry
예약자명	strawberry
호텔명	제주그랜드호텔
객실 타입	--객실선택--
입실 일자	연도-월-일
퇴실 일자	연도-월-일
투숙 인원	--인원선택-- (기준 2인/최대 3인, 기준 인원 초과시 추가비용 발생, 스탠다드-25000원, 프리미엄-70000원)
결제 금액	0원

예약 변경 변경 취소

유효성 체크 후

```
<form action="reserv_update.jsp" onsubmit="return check();" >
<!-- a태그로 받아온 예약번호 값을 form에 같이 넘겨주는 방법 -->
<input type="hidden" name="reserv_num" value="<%= reserv_num2 %>" >
```

폼 안의 값들을
reserv_update로 전송

```
<script type="text/javascript">
<%
if(result != 0){ %>
    alert('예약정보 수정을 성공했습니다.')
    location.href="reserv_list.jsp"
<%}else{%>
    alert('예약정보 실패했습니다. 다시 시도해주세요.')
<%}
%>
</script>
```

localhost:8887 내용:
예약정보 수정을 성공했습니다.

확인

```
<%
//form action으로 넘어온 값들
//예약번호로 특정 레코드값을 수정할 것이므로 reserv_up에서 예약번호 값을 넘겨줘야 한다.
//예약번호 값을 넘겨주는 매개체는 <input type="hidden"> 이 작업이 없을 시 null로, 에러 발생
String reserv_num = request.getParameter("reserv_num");
int reserv_num2 = Integer.parseInt(reserv_num);
//변경한 예약정보 값
String room_sort = request.getParameter("room_sort");
String check_in = request.getParameter("check_in");
String check_out = request.getParameter("check_out");
String reserv_count = request.getParameter("reserv_count");
int reserv_count2 = Integer.parseInt(reserv_count);
String reserv_price = request.getParameter("reserv_price"); //00000원
//금액(숫자)+원"인 값에서 숫자만 추출
String reserv_price2 = reserv_price.substring(0, reserv_price.length() - 1);
int reserv_price3 = Integer.parseInt(reserv_price2);
//예약번호를 조건으로 update메서드를 사용할 것이므로, 예약번호 값을 dto에 실는다.
ReservDTO dto = new ReservDTO();
dto.setReserv_num(reserv_num2);
dto.setRoom_sort(room_sort);
dto.setCheck_in(check_in);
dto.setCheck_out(check_out);
dto.setReserv_count(reserv_count2);
dto.setReserv_price(reserv_price3);
//예약번호와 변경된 입력값을 담은 dto를 update메서드에 전송
ReservDAO dao = new ReservDAO();
int result = dao.update(dto);
%>
```

reservDAO.java

```
public int update(ReservDTO dto) {
```

```
// 3. sql문을 만든다.
String sql = "update reservation set room_sort = ?, check in = ?,
check_out = ?, reserv_count = ?, reserv_price = ? where reserv_num = ?";
PreparedStatement ps = con.prepareStatement(sql);
ps.setString(1, dto.getRoom_sort());
ps.setString(2, dto.getCheck_in());
ps.setString(3, dto.getCheck_out());
ps.setInt(4, dto.getReserv_count());
ps.setInt(5, dto.getReserv_price());
ps.setInt(6, dto.getReserv_num());
```

- 예약취소

전체 예약된 객실의 개수: 2개

예약 번호	예약자명	호텔명	객실 타입	입실 일자	퇴실 일자	투숙 인원	결제 금액	
98	strawberry	제주센트럴시티호텔	스탠다드	2021-09-05	2021-09-06	1	30000	예약변경 예약취소
99	strawberry	제주그랜드호텔	프리미엄		-15	3		

localhost:8887 내용:

예약이 취소되었습니다.

확인

reserv_list.jsp

```
<a href="reserv_delete.jsp?reserv_num=<%=dto6.getReserv_num()%>">
<button style="width:100px">예약취소</button></a>
```

reserv_delete.jsp

```
//a태그로 받아온 예약번호 값으로 등록된 예약 내역 지우기
String reserv_num = request.getParameter("reserv_num");
int reserv_num2 = Integer.parseInt(reserv_num);
//취소하고 싶은 글의 예약번호 값을 dto에 실는다.
ReservDTO dto = new ReservDTO();
dto.setReserv_num(reserv_num2);
//delete메서드에 dto를 입력값으로 전송
ReservDAO dao = new ReservDAO();
int result = dao.delete(dto);
```

DB처리한 결과
result를 반환예약 취소 성공
여부 팝업

```
<script type="text/javascript">
<%
if(result != 0){ %>
    alert('예약이 취소되었습니다.')
    location.href="reserv_list.jsp"
<%}else{%>
    alert('예약 취소를 실패했습니다. 다시 시도해주세요.')
<%}
%>
</script>
```

예약 내역
페이지로 이동

```
public int delete(ReservDTO dto) {
// 3. sql문을 만든다.
String sql = "delete from reservation where reserv_num = ?";
PreparedStatement ps = con.prepareStatement(sql);
ps.setInt(1, dto.getReserv_num());
return result;
```

reservDAO.java